

# Лабораторная работа №7 по предмету Операционные системы

Группа НПМбв-02-19

Воронцов Павел Васильевич

## Содержание

### 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, начинающиеся с символа `c`? 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`. Можно ли определить этот идентификатор более простым способом?
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, имеющихся в вашем домашнем каталоге.

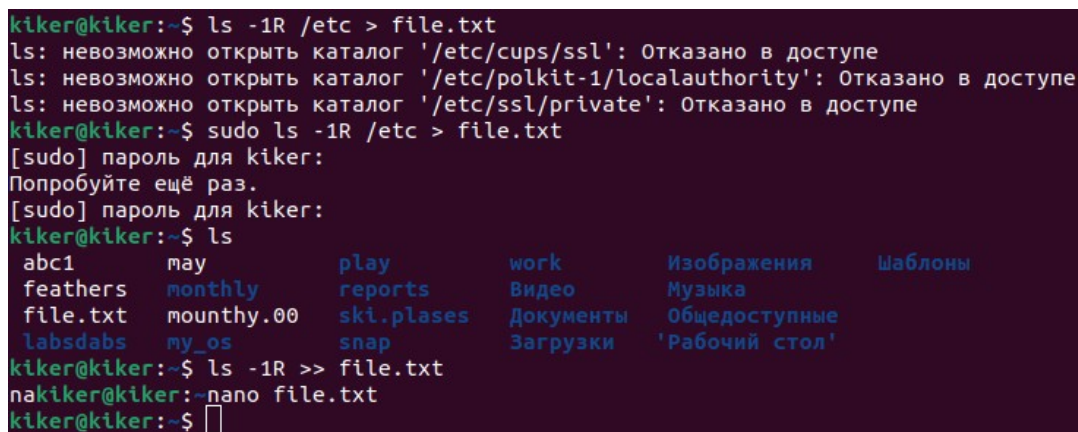
## Выполнение лабораторной работы

Запишите в файл `file.txt` названия файлов, содержащихся в каталоге `/etc`. Допишите в этот же файл названия файлов, содержащихся в вашем домашнем каталоге.

Для записи `/etc` в `file.txt` и файлов домашнего каталога (рис.1):

```
sudo ls -lR /etc > file.txt
ls -lR >> file.txt
```

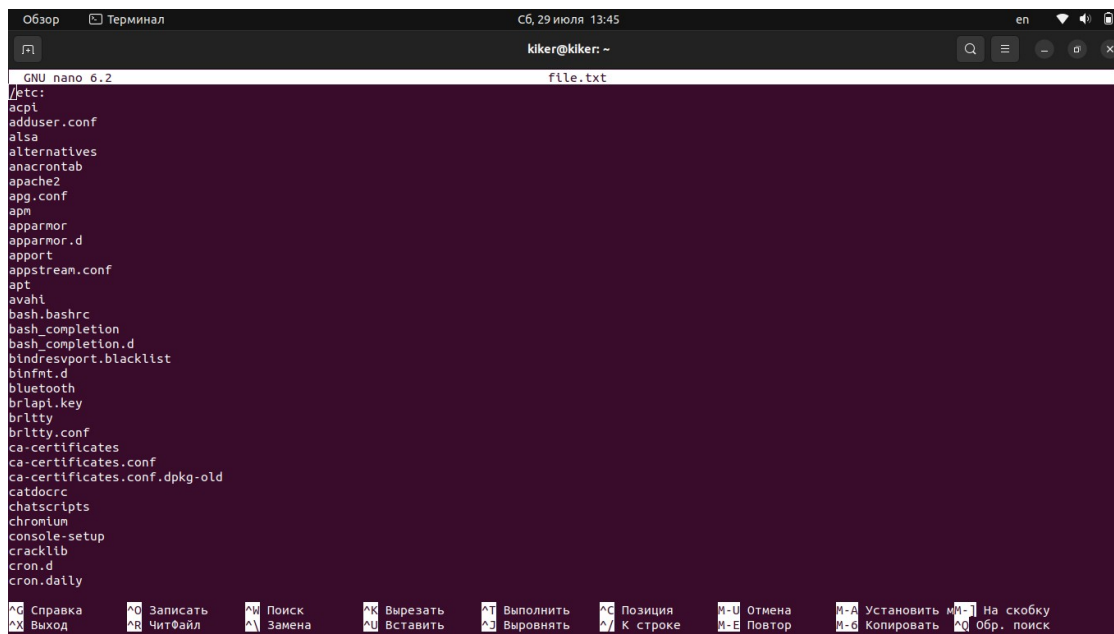
`sudo` нужно для первой команды, так как без нее в `/etc` не дает выполнить команду.



```
kiker@kiker:~$ ls -lR /etc > file.txt
ls: невозможно открыть каталог '/etc/cups/ssl': Отказано в доступе
ls: невозможно открыть каталог '/etc/polkit-1/localauthority': Отказано в доступе
ls: невозможно открыть каталог '/etc/ssl/private': Отказано в доступе
kiker@kiker:~$ sudo ls -lR /etc > file.txt
[sudo] пароль для kiker:
Попробуйте ещё раз.
[sudo] пароль для kiker:
kiker@kiker:~$ ls
abc1      may      play     work     Изображения  Шаблоны
feathers  monthly  reports  Видео    Музыка
file.txt  mounthy.00 ski.plases  Документы  Общедоступные
labsdabs  my_os    snap     Загрузки  'Рабочий стол'
kiker@kiker:~$ ls -lR >> file.txt
kiker@kiker:~$ nano file.txt
kiker@kiker:~$
```

Рисунок 1

На рис.2 видно, где в созданном файле заканчивается перечисление файлов и подкаталогов `/etc` и начинается перечисление файлов и подкаталогов домашнего каталога.



## Рисунок 2

Выведите имена всех файлов из file.txt, имеющих расширение .conf, после чего запишите их в новый текстовый файл conf.txt.

Использую grep для этого (рис.3).

После grep указана маска для поиска в file.txt файлов с расширением .conf.

```
grep '.conf$' file.txt > conf.txt
```

```
kiker@kiker:~$ grep '.conf$' file.txt > conf.txt
kiker@kiker:~$ cat conf.txt
adduser.conf
apg.conf
appstream.conf
brltty.conf
ca-certificates.conf
dconf
debconf.conf
deluser.conf
e2scrub.conf
fprintd.conf
fuse.conf
gai.conf
hdparm.conf
host.conf
kernel-img.conf
kerneloops.conf
ld.so.conf
libao.conf
libaudit.conf
logrotate.conf
mke2fs.conf
nftables.conf
nsswitch.conf
pam.conf
pnm2ppa.conf
resolv.conf
rsyslog.conf
rygel.conf
sensors3.conf
smartd.conf
sudo.conf
sudo_logsrvd.conf
sysctl.conf
xfs.conf
```

Рисунок 3

Определите, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа с? Предложите несколько вариантов, как это сделать.

Я использовала три команды (рис.4-6):

```
ls -d c* (1)
find . -type f -name 'c*' (2)
ls | grep '^c*' (3)
```

```
kiker@kiker:~$ ls -d c*
conf.txt
kiker@kiker:~$
```

Рисунок 4

1: выведет файлы в текущем каталоге начинающиеся с 'с'

```
kiker@kiker:~$ find . -type f -name 'c*'
././Scilab/scilab-branch-6.1/configuration.xml
././local/share/evolution/addressbook/system/contacts.db
././local/share/evolution/calendar/system/calendar.ics
././local/share/flatpak/repo/config
././var/app/org.gnome.gitlab.somas.Apostrophe/cache/fontconfig/cabbd14511b9e8a55e92af97fb3a0461-le64.cache-8
././var/app/org.gnome.gitlab.somas.Apostrophe/cache/fontconfig/c3bb64600bf7a4aa73b55dc8bb82b27b-le64.cache-8
././var/app/org.gnome.gitlab.somas.Apostrophe/cache/fontconfig/c855463f699352c367813e37f3f70ea7-le64.cache-8
././var/app/org.gnome.gitlab.somas.Apostrophe/cache/fontconfig/c5c45a61289222e0d30b1a26ef4effbe-le64.cache-8
././cache/mesa_shader_cache/0c/c451c057ae0b244491501ef885723c409ea693
././cache/mesa_shader_cache/0f/c336cc22730a0cb17e205c328908665487a624
././cache/mesa_shader_cache/82/cef79949daf5cd35ab8f0812fdda61e0a00a38
././cache/mesa_shader_cache/d3/cec7baf7871917ba978f5a75ada9b14e65d84f
././cache/mesa_shader_cache/cf/c45bcc88ad0313f90cb3630791709fdb51d39
././cache/mesa_shader_cache/94/cb0a9c916280c9a30d680ddcf7bc3dc0f55687
././cache/mesa_shader_cache/ca/c5af70b7a494da826ca1b416d4b23ea72fefe7
././cache/mesa_shader_cache/c7/cf01a00403dae4cd5a7482674172b52c27bf51
././cache/mesa_shader_cache/3d/c98713d78d5835beb14efcaf3a9ce2d3cce896
././cache/mesa_shader_cache/e2/cfeca8ed772dfba9b08b9fb13573a6bd25aa4f
././cache/mesa_shader_cache/ed/cd4d16fdab8d9aa72ef0998e1a4462e57dc061
././cache/mesa_shader_cache/a3/ce9b543568e33a0a020940e8d20fd4a5781878
././cache/mesa_shader_cache/72/cc156000272bd6a57ca982e9fe824a428d9406
././cache/mesa_shader_cache/d1/cdd05145e534dbea12083546cdf390c72faa3d
././cache/mesa_shader_cache/1b/c8c52bf35f8debef80a6ec3cf8058462d4a529
././cache/mesa_shader_cache/86/c372b60644fbf3d1720b59e8045bd8136b221a
././cache/mesa_shader_cache/fd/cf1781afb62b5698749967768cef02ab0360db
././cache/mesa_shader_cache/c2/c548c8d0e7823f536eb66efd3d9f3904358c50
././cache/mesa_shader_cache/0d/cf7515d9f0fa25b54ce6ae9d37e67585657f6b
././cache/mesa_shader_cache/f4/c5dd8c6e46140a3c42aeeebe555f64d202424d
././cache/mesa_shader_cache/06/c0d7fd8e7356086ce7c0a29fe296df731c42dc
././cache/thumbnails/large/c06772face77e2370b6d2ee60af0121a.png
././cache/thumbnails/large/ca78f778b8df7e97e652edc7712714b0.png
././cache/thumbnails/large/cddb7348e00ff1c7eb0eca0af63cbe5d.png
././cache/thumbnails/large/cc023c56d39900d636b776c69039ab13.png
././cache/thumbnails/large/cb87aa47f8ea5fb93b8ed6e6afe4562c.png
././cache/thumbnails/large/c9d5537a86b1af8921137cdf92d2670c.png
././cache/thumbnails/large/cb175a3b484f04164bfce6fe0db6ac54.png
././cache/thumbnails/large/c42d8b48603e64b536d211a82f6eee5f.png
```

Рисунок 5

2: выведет полный путь к файлам из всех подкаталогов текущего каталога, начинающимся с 'с'

```
kiker@kiker:~$ ls | grep '^c*'
abc1
conf.txt
feathers
file.txt
labsdabs
may
monthly
mounthy.00
my_os
play
reports
ski.plases
snap
work
Видео
Документы
Загрузки
Изображения
Музыка
Общедоступные
Рабочий стол
Шаблоны
kiker@kiker:~$
```

Рисунок 6

З: выведет список файлов и подкаталогов текущего каталога и подсветит нужные файлы

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

Использую команду:

```
find /etc -name 'h*' | more -s
```

Здесь more это команда постраничного вывода.

Но моя команда выводит не только названия файлов в каталоге /etc, но и файлы из подкаталогов /etc (рис.7) и их пути

```
kiker@kiker:~$ find /etc -name 'h*' | more -s
find: '/etc/ssl/private': Отказано в доступе
find: '/etc/cups/ssl': Отказано в доступе
find: '/etc/polkit-1/localauthority': Отказано в доступе
/etc/hdparm.conf
/etc/brltty/Input/hm
/etc/brltty/Input/hw
/etc/brltty/Input/bm/horizontal.kti
/etc/brltty/Input/hd
/etc/brltty/Input/ht
/etc/brltty/Contraction/ha.ctb
/etc/brltty/Text/hi.ttb
/etc/brltty/Text/he.ttb
/etc/brltty/Text/hy.ttb
/etc/brltty/Text/hu.ttb
/etc/brltty/Text/hr.ttb
/etc/host.conf
/etc/hp
/etc/hp/hplip.conf
/etc/ubuntu-advantage/help_data.yaml
/etc/initramfs-tools/hooks
/etc/hosts.deny
/etc/apparmor.d/abstractions/hosts_access
/etc/apparmor.d/tunables/home.d
/etc/apparmor.d/tunables/home
/etc/hosts.allow
/etc/hostid
/etc/kernel/header_postinst.d
/etc/hostname
/etc/avahi/hosts
/etc/hosts
/etc/sane.d/hs2p.conf
/etc/sane.d/hp.conf
/etc/sane.d/hp3900.conf
/etc/sane.d/hp4200.conf
/etc/sane.d/hp5400.conf
/etc/sane.d/hpsj5s.conf
```

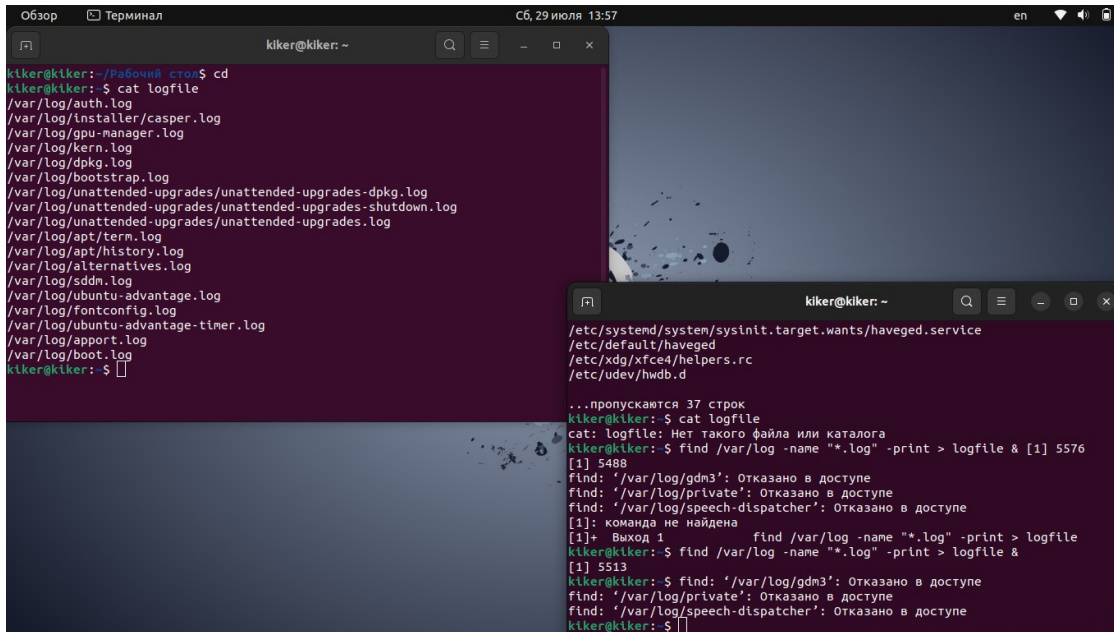
Рисунок 7

Запустите в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log (рис.8)

На рис. 8 и терминал с командой, и просмотр logfile.

```
find /var/log -name "*.log" -print > logfile &
```





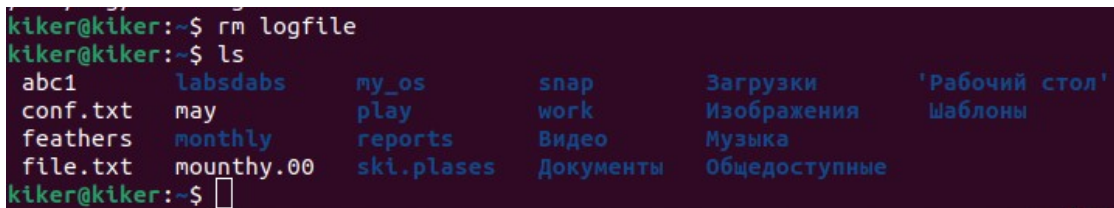
```
kikergtkiker:~/Рабочий стол$ cd
kikergtkiker:~$ cat logfile
/var/log/auth.log
/var/log/installer/casper.log
/var/log/gpu-manager.log
/var/log/kern.log
/var/log/dpkg.log
/var/log/bootstrap.log
/var/log/unattended-upgrades/unattended-upgrades-dpkg.log
/var/log/unattended-upgrades/unattended-upgrades-shutdown.log
/var/log/unattended-upgrades/unattended-upgrades.log
/var/log/apt/term.log
/var/log/apt/history.log
/var/log/alternatives.log
/var/log/sddm.log
/var/log/ubuntu-advantage.log
/var/log/fontconfig.log
/var/log/ubuntu-advantage-timer.log
/var/log/apport.log
/var/log/boot.log
kikergtkiker:~$

/etc/systemd/system/sysinit.target.wants/haveged.service
/etc/default/haveged
/etc/xdg/xfce4/helpers.rc
/etc/udev/hwdb.d

...пропускаются 37 строк
kikergtkiker:~$ cat logfile
cat: logfile: Нет такого файла или каталога
kikergtkiker:~$ find /var/log -name "*.log" -print > logfile & [1] 5576
[1] 5488
find: '/var/log/gdm3': Отказано в доступе
find: '/var/log/private': Отказано в доступе
find: '/var/log/speech-dispatcher': Отказано в доступе
[1]: команда не найдена
[1]+ Выход 1 find /var/log -name "*.log" -print > logfile
kikergtkiker:~$ find /var/log -name "*.log" -print > logfile &
[1] 5513
kikergtkiker:~$ find: '/var/log/gdm3': Отказано в доступе
find: '/var/log/private': Отказано в доступе
find: '/var/log/speech-dispatcher': Отказано в доступе
kikergtkiker:~$
```

Рисунок 8

Удалите файл ~/logfile (рис.9)



```
kikergtkiker:~$ rm logfile
kikergtkiker:~$ ls
abc1      labsdabs  my_os     snap      Загрузки  'Рабочий стол'
conf.txt  may       play      work      Изображения  Шаблоны
feathers  monthly  reports   Видео     Музыка
file.txt  mounthy.00 ski.plases Документы  Общедоступные
kikergtkiker:~$
```

Рисунок 9

Запустите из консоли в фоновом режиме редактор gedit (рис.10)

На рис. 10 и запуск команды в консоли gedit &, и открывшееся окно текстовой редактора.



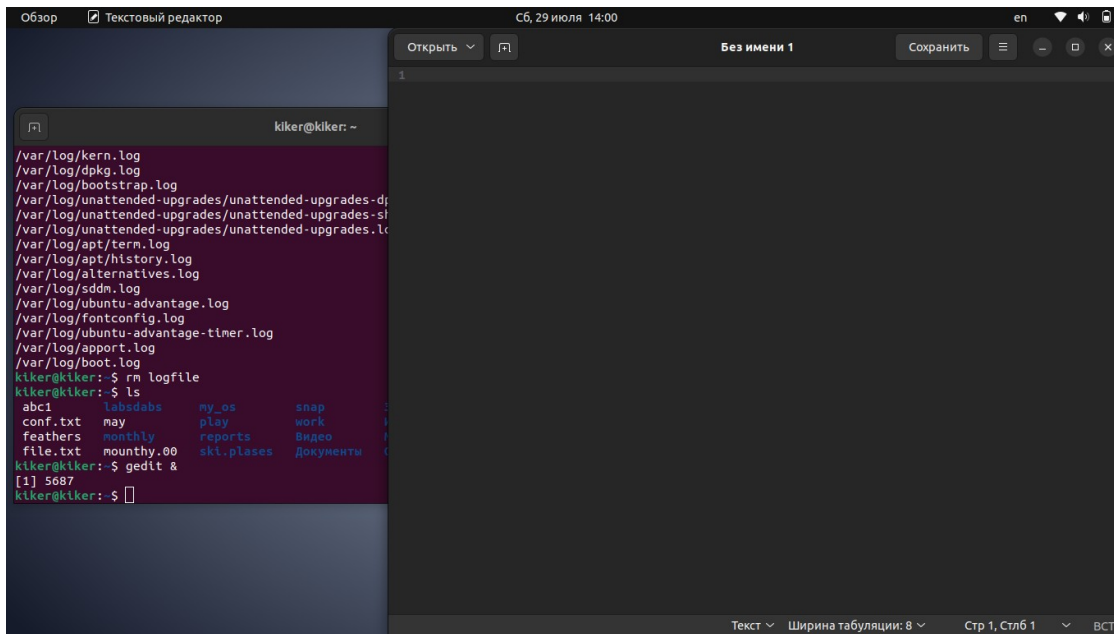


Рисунок 10

Определите идентификатор процесса `gedit`, используя команду `ps`, конвейер и фильтр `grep`. Можно ли определить этот идентификатор более простым способом? (рис. 11)

Можно либо найти из списка после вызова команды `ps aux`, либо отфильтровать еще с помощью `grep` по имени процесса (рис.11). Еще проще вряд ли можно.

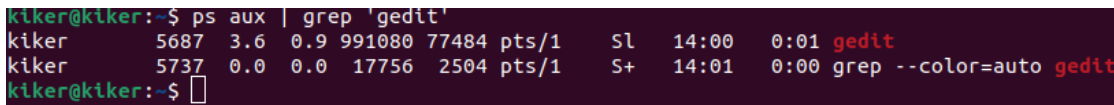


Рисунок 11

Прочтите справку (`man`) команды `kill`, после чего используйте её для завершения процесса `gedit` (рис.12)

`kill` PID, где PID это номер процесса, который мы нашли в прошлом пункте.



Рисунок 12

Выполните команды `df` и `du`, предварительно получив более подробную информацию об этих командах, с помощью команды `man` (рис. 13-14)

df показывает информацию о файловых системах, сколько занято пространства в них, и о точках монтирования (рис.13)

```
kiker@kiker:~$ df
```

Файл. система	1K-блоков	Использовано	Доступно	Использовано%	Смонтировано в
tmpfs	799052	2024	797028	1%	/run
/dev/sda2	479079112	29590400	425079320	7%	/
tmpfs	3995256	0	3995256	0%	/dev/shm
tmpfs	5120	4	5116	1%	/run/lock
/dev/sda1	523244	6216	517028	2%	/boot/efi
tmpfs	799048	208	798840	1%	/run/user/1000

Рисунок 13

du показывает информацию о занятом пространстве в байтах (рис.14)

```
kiker@kiker:~$ du | sort -n -r
```

6529132	.
5514364	./snap
4544992	./snap/steam
4544972	./snap/steam/common
4541628	./snap/steam/common/.local
4541624	./snap/steam/common/.local/share
4541512	./snap/steam/common/.local/share/Steam
2888004	./snap/steam/common/.local/share/Steam/steamapps
2880668	./snap/steam/common/.local/share/Steam/steamapps/common
2880664	./snap/steam/common/.local/share/Steam/steamapps/common/Underlords
2880660	./snap/steam/common/.local/share/Steam/steamapps/common/Underlords/game
2504708	./snap/steam/common/.local/share/Steam/steamapps/common/Underlords/game/dac
874844	./Заргузки
840784	./snap/firefox
840604	./snap/firefox/common
786964	./snap/firefox/common/.cache
783936	./snap/firefox/common/.cache/mozilla
783932	./snap/firefox/common/.cache/mozilla/firefox
783928	./snap/firefox/common/.cache/mozilla/firefox/7lbsnx9x.default
755148	./snap/firefox/common/.cache/mozilla/firefox/7lbsnx9x.default/cache2
754212	./snap/firefox/common/.cache/mozilla/firefox/7lbsnx9x.default/cache2/entries
549120	./Заргузки/scilab-6.1.1.bin.linux-x86_64
549116	./Заргузки/scilab-6.1.1.bin.linux-x86_64/scilab-6.1.1
533980	./snap/steam/common/.local/share/Steam/ubuntu12_32
366316	./snap/steam/common/.local/share/Steam/package
299568	./Заргузки/scilab-6.1.1.bin.linux-x86_64/scilab-6.1.1/share
292796	./snap/steam/common/.local/share/Steam/ubuntu12_64
292132	./Заргузки/scilab-6.1.1.bin.linux-x86_64/scilab-6.1.1/share/scilab
291212	./Заргузки/scilab-6.1.1.bin.linux-x86_64/scilab-6.1.1/share/scilab/modules
252932	./snap/steam/common/.local/share/Steam/steamapps/common/Underlords/game/dac_desktop
225916	./snap/steam/common/.local/share/Steam/ubuntu12_32/steam-runtime
188232	./snap/steam/common/.local/share/Steam/ubuntu12_32/steam-runtime/usr
169572	./Заргузки/scilab-6.1.1.bin.linux-x86_64/scilab-6.1.1/thirdparty
160920	./snap/steam/common/.local/share/Steam/ubuntu12_32/steam-runtime/usr/lib
133960	./snap/steam/common/.local/share/Steam/tenfoot
133956	./snap/steam/common/.local/share/Steam/tenfoot/resource

Рисунок 14

Воспользовавшись справкой команды find, выведите имена всех директорий, имеющих в вашем домашнем каталоге (рис.15)

-type d – тип для вывода - директория (каталог)

-maxdepth 1 – глубина вывода содержимого, в данном случае выведет только подкаталоги и не будет выводить то, что внутри подкаталогов

```
find -type d -maxdepth 1
```

```
kikergtkker:~$ find -type d -maxdepth 1
find: warning: you have specified the global option -maxdepth after the argument -type, but global options are not positional, i.e., -maxdepth affects
tests specified before it as well as those specified after it. Please specify global options before other arguments.
./
./Scilab
./Музыка
./monthly
./ssh
./my_os
./reports
./local
./gnupg
./var
./ski.places
./labsdabs
./cache
./Видео
./Изображения
./Документы
./Загрузки
./Рабочий стол
./Диагностика
./Общедоступные
./snap
./config
./play
./work
kikergtkker:~$
```

Рисунок 15

## Выводы

В ходе выполнения данной лабораторной работы я ознакомилась с инструментами поиска файлов и фильтрации текстовых файлов, приобрела практические навыки по управлению процессам и проверке использования диска и файловых систем.

## Контрольные вопросы

1. Какие потоки ввода-вывода вы знаете?  
В Linux есть три стандартных потока ввода-вывода: stdin (стандартный поток ввода), stdout (стандартный поток вывода) и stderr (стандартный поток ошибок).
2. Объясните разницу между операцией `>` и `>>`.  
Операция `>` используется для перенаправления вывода команды в файл, заменяя содержимое файла новым выводом. Операция `>>` также перенаправляет вывод в файл, но не заменяет его содержимое, а дописывает новый вывод в конец файла.
3. Что такое конвейер?  
Конвейер (pipeline) в Linux - это последовательность команд, в которой вывод одной команды используется в качестве ввода следующей команды, а также для передачи вывода из одного процесса в другой.
4. Что такое процесс? Чем это понятие отличается от программы?  
Процесс в операционной системе Linux - это экземпляр выполняющейся программы, который обладает собственными ресурсами, такими как память, файловые дескрипторы и стек вызовов. Программа же представляет собой набор инструкций, которые могут быть запущены в процессе выполнения.

5. Что такое PID и GID?  
PID (Process ID) - это уникальный числовой идентификатор, который присваивается каждому процессу в Linux. GID (Group ID) - это идентификатор группы, к которой относится процесс.
6. Что такое задачи и какая команда позволяет ими управлять?  
Задача в Linux - это единица работы, которую нужно выполнить в определенный момент времени. Команда для управления задачами - это "cron", который позволяет создавать и настраивать периодические задания в системе.
7. Найдите информацию об утилитах top и htop. Каковы их функции?  
Top и htop - это утилиты мониторинга процессов в Linux. Они отображают список запущенных процессов и позволяют отслеживать использование ресурсов системы, таких как центральный процессор (CPU) и оперативная память (RAM).
8. Назовите и дайте характеристику команде поиска файлов.  
Приведите примеры использования этой команды.  
Команда поиска файлов в Linux называется "find". Она позволяет искать файлы и директории в файловой системе по различным параметрам, таким как имя, тип, размер, дата изменения и т.д.  
Примеры использования команды find:  
Найти все файлы с расширением ".txt" в директории /home/user: `find /home/user -name "*.txt"`  
Найти все файлы больше 10 МБ в директории /var/log: `find /var/log -size +10M`
9. Можно ли по контексту (содержанию) найти файл? Если да, то как?  
Да, можно искать файлы по содержанию. Для этого используется команда "grep". Например, чтобы найти все файлы, содержащие слово "error", в директории /var/log, нужно выполнить следующую команду: `grep -r "error" /var/log`
10. Как определить объем свободной памяти на жёстком диске?  
Чтобы определить объем свободной памяти на жестком диске в Linux, можно использовать команду "df". Она отображает информацию о доступном месте на файловых системах. Команда df без параметров покажет информацию по всем файловым системам. Например: `df -h`.
11. Как определить объем вашего домашнего каталога?  
Чтобы определить объем своего домашнего каталога в Linux, можно использовать команду "du". Она позволяет посчитать размер файлов и директорий в указанной директории. Чтобы посчитать размер домашней директории, нужно выполнить следующую команду: `du -sh ~/`

12. Как удалить зависший процесс?

Чтобы удалить зависший процесс в Linux, можно воспользоваться командой "kill". Сначала нужно определить PID с помощью команды ps aux.