

Отчёт по лабораторной работе №6

Операционные системы

Коннова Татьяна Алексеевна

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Контрольные вопросы	18
5	Выводы	22

Список иллюстраций

3.1	Запись файлов в file.txt и проверка	7
3.2	Запись файлов с расширением .conf в новый текстовый файл. Выполнение проверки	8
3.3	Определение файлов в домашнем каталоге, начинавшихся с символа с	9
3.4	Команда для вывода на экран файлов	10
3.5	Запись файлов в logfile	11
3.6	Удаление файла logfile и проверка	11
3.7	Запуск редактора gedit в фоновом режиме	12
3.8	Определение идентификатора процесса gedit	13
3.9	man kill	14
3.10	Завершение процесса gedit с помощью kill	14
3.11	man df	15
3.12	man du	15
3.13	Выполнение команд df и du	16
3.14	man find	16
3.15	Вывод имён всех директорий в домашнем каталоге	17

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

Запишем в файл file.txt название файлов, содержащихся в каталоге /etc. Далее допишем в этот же файл название файлов, содержащихся в домашнем каталоге. Выполним проаерку, прочитав этот файл с помощью команды: cat (рис. [3.1]).

```
[takonnova@fedora stagel]$  
[takonnova@fedora stagel]$ ls -a /etc > file.txt  
[takonnova@fedora stagel]$ ls -a ~ >> file.txt  
[takonnova@fedora stagel]$ 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
```

Рис. 3.1: Запись файлов в file.txt и проверка

Выведем имена всех файлов из file.txt, имеющие расширение .conf, после чего

запишем их в новый текстовый файл conf.txt. Выполним проверку, прочитав новый файл. (рис. [3.2]).

```
[takonnova@fedora stage1]$  
[takonnova@fedora stage1]$ grep -e '\.conf$' file.txt > conf.txt  
[takonnova@fedora stage1]$ cat conf.txt  
anthy-unicode.conf  
appstream.conf  
asound.conf  
brltty.conf  
chrony.conf  
dleyna-renderer-service.conf  
dleyna-server-service.conf  
dnsmasq.conf  
dracut.conf  
extlinux.conf  
fprintd.conf  
fuse.conf  
host.conf  
idmapd.conf  
jwhois.conf  
kdump.conf  
krb5.conf  
ld.so.conf  
libaudit.conf  
libuser.conf  
locale.conf  
logrotate.conf  
man_db.conf  
mke2fs.conf  
mtools.conf  
nfs.conf  
nfsmount.conf  
nsswitch.conf  
opensc.conf  
opensc-x86_64.conf  
passwdqc.conf  
request-key.conf
```

Рис. 3.2: Запись файлов с расширением .conf в новый текстовый файл. Выполнение проверки

Определим, какие файлы в домашнем каталоге имеют имена, начинавшиеся с символа с? Приведём 2 варианта выполнения этой задачи `find ~ -maxdepth 1 -name "с*" -print /home/takonnova/conf.txt` (рис. [3.3]).


```
[takonnova@fedora ~]$ ls -a ~ | grep c*  
conf.txt  
[takonnova@fedora ~]$ find ~ -maxdepth 1 -name "c*" -print  
/home/takonnova/conf.txt
```

Рис. 3.3: Определение файлов в домашнем каталоге, начинавшихся с символа с

```
find /etc -maxdepth 1 -name "h*" -print | less
```

Выведем на экран (по странично) имена файлов из каталога /etc, начинающихся с символа h (рис. [3.4]).

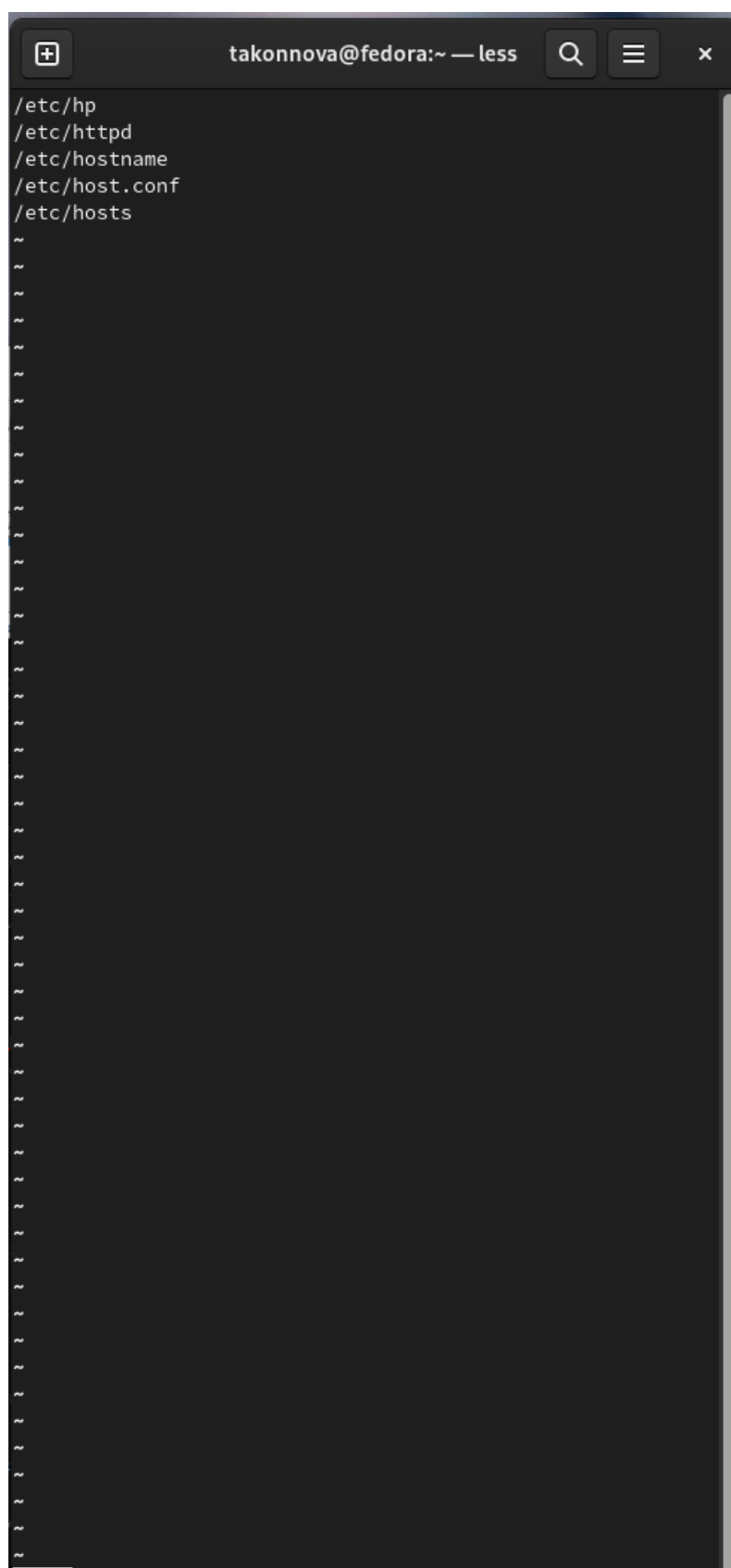


Рис. 3.4: Команда для вывода на экран файлов

Запустим в фоновом режиме процесс, который будет записывать в файл `~/logfile` файлы, имена которых начинаются с `log` (рис. [3.5]).

```
takonnova@fedora:~  
[takonnova@fedora ~]$ find / -name "log." > logfile &  
[1] 44828  
[takonnova@fedora ~]$ find: '/boot/lost+found': Отказано в доступе  
find: '/boot/loader/entries': 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
```

Рис. 3.5: Запись файлов в logfile

Удалим файл `~/logfile` командой `rm` и выполним проверку (рис. [3.6]).

[illegible]

Рис. 3.6: Уделение файла logfile и проверка

Запустим из консоли в фоновом режиме редактор gedit. (рис. [3.7]).

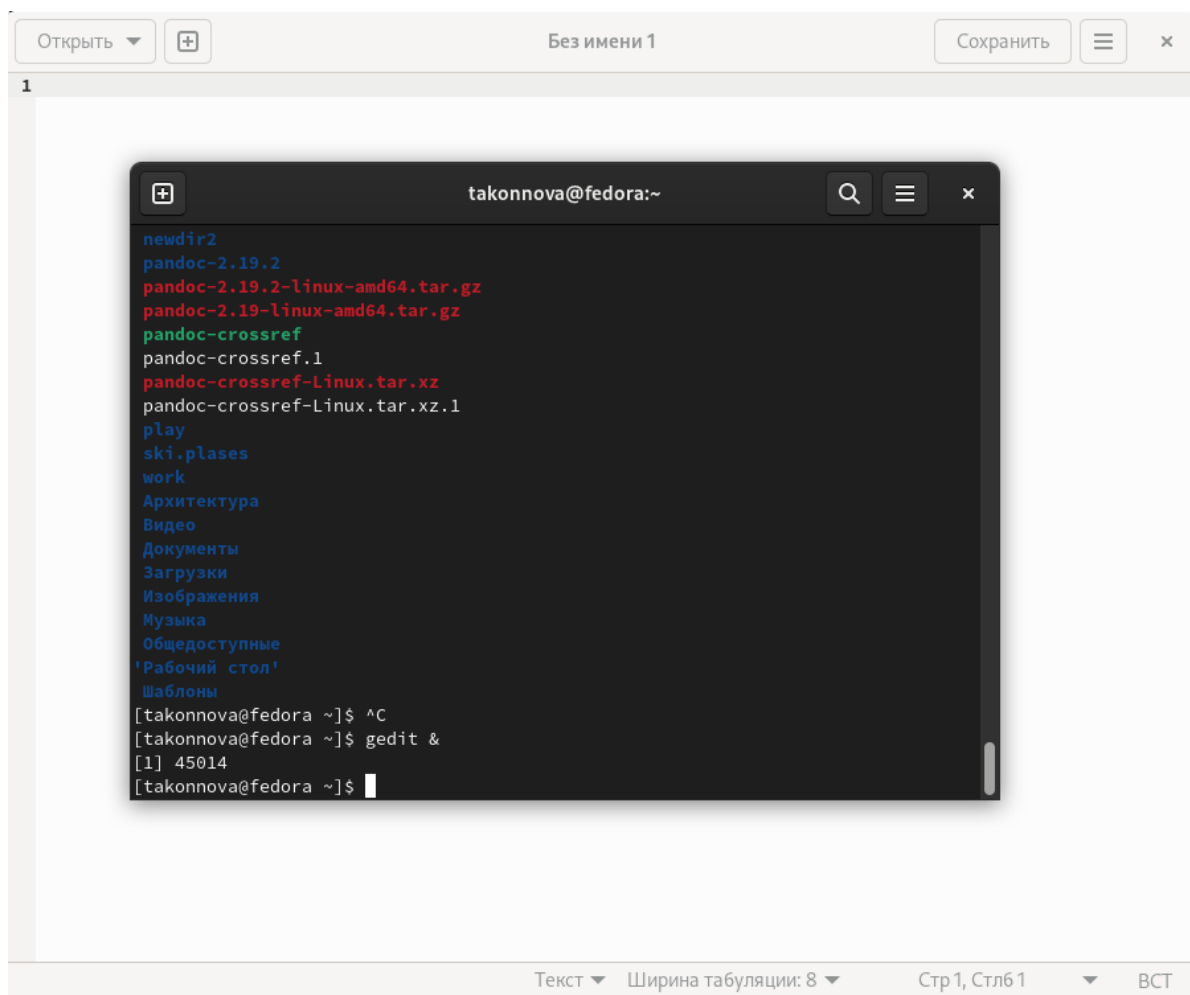
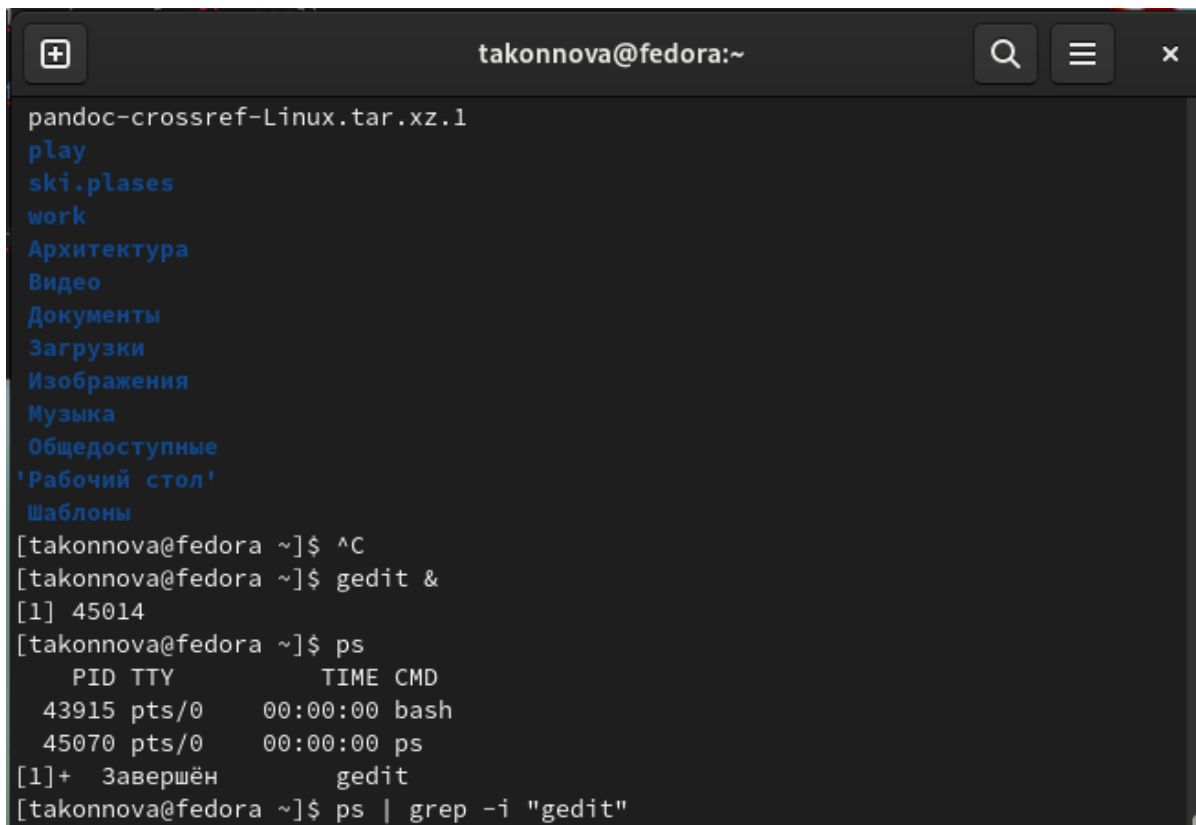


Рис. 3.7: Запуск редактора gedit в фоновом режиме

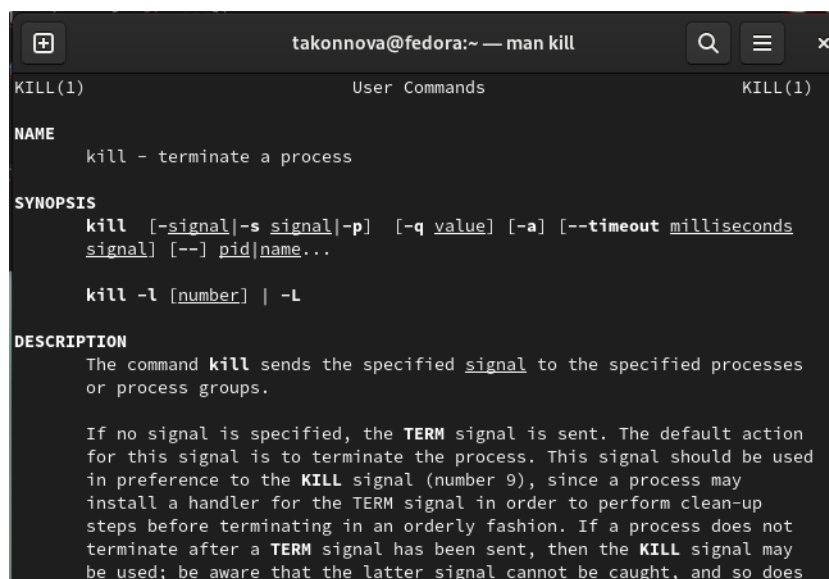
Определим идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep (рис. [3.8]).

A terminal window titled 'takonnova@fedora:~' with search, menu, and close icons in the title bar. The terminal shows a file manager sidebar on the left with items like 'pandoc-crossref-Linux.tar.xz.1', 'play', 'ski.places', 'work', 'Архитектура', 'Видео', 'Документы', 'Загрузки', 'Изображения', 'Музыка', 'Общедоступные', 'Рабочий стол', and 'Шаблоны'. The main terminal area shows the following commands and output:

```
[takonnova@fedora ~]$ ^C
[takonnova@fedora ~]$ gedit &
[1] 45014
[takonnova@fedora ~]$ ps
  PID TTY          TIME CMD
  43915 pts/0    00:00:00 bash
  45070 pts/0    00:00:00 ps
[1]+  Завершён      gedit
[takonnova@fedora ~]$ ps | grep -i "gedit"
```

Рис. 3.8: Определение идентификатора процесса gedit

Прочитаем справку (man) команды kill (рис. [3.9]), после чего используем её для завершения процесса gedit (рис. [3.10]).



```
takonnova@fedora:~ — man kill
KILL(1) User Commands KILL(1)

NAME
    kill - terminate a process

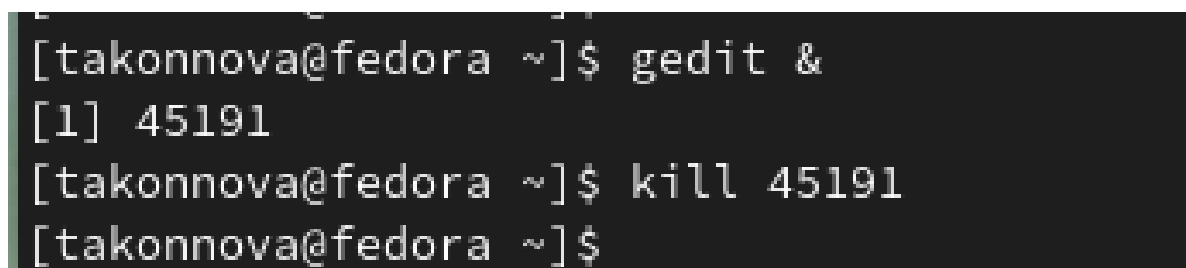
SYNOPSIS
    kill [-signal|-s signal|-p] [-q value] [-a] [--timeout milliseconds
    signal] [--] pid|name...

    kill -l [number] | -L

DESCRIPTION
    The command kill sends the specified signal to the specified processes
    or process groups.

    If no signal is specified, the TERM signal is sent. The default action
    for this signal is to terminate the process. This signal should be used
    in preference to the KILL signal (number 9), since a process may
    install a handler for the TERM signal in order to perform clean-up
    steps before terminating in an orderly fashion. If a process does not
    terminate after a TERM signal has been sent, then the KILL signal may
    be used; be aware that the latter signal cannot be caught, and so does
```

Рис. 3.9: man kill



```
[takonnova@fedora ~]$ gedit &
[1] 45191
[takonnova@fedora ~]$ kill 45191
[takonnova@fedora ~]$
```

Рис. 3.10: Завершение процесса gedit с помощью kill

Выполним команды `df` и `du` (рис. [3.13]), предварительно получив более подробную информацию об этих командах, с помощью команды `man`.

Команда `df` показывает размер каждого смонтированного раздела диска (рис. [3.11]).

```
takonnova@fedora:~ — man DF
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 cur-
    rently 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 ver-
    sion of df cannot show the space available on unmounted file systems,
    because on most kinds of systems doing so requires very nonportable in-
    timate knowledge of file system structures.

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

Рис. 3.11: man df

Команда du показывает число килобайт, используемое каждым файлом или каталогом (рис. [3.12]).

```
takonnova@fedora:~ — man DU
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 directo-
    ries.

    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

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

Рис. 3.12: man du

```
takonnova@fedora:~  
[takonnova@fedora ~]$ df  
[takonnova@fedora ~]$ df  
Файловая система 1К-блоков  Использовано  Доступно  Использовано%  Смонтировано в  
devtmpfs            4096          0      4096           0% /dev  
tmpfs                2006724      24032   1982692          2% /dev/shm  
tmpfs                802692       1460    801232          1% /run  
/dev/sda2            51379200    15376732 35542180        31% /  
/dev/sda2            51379200    15376732 35542180        31% /home  
tmpfs                2006724         56   2006668          1% /tmp  
/dev/sda1            996780      229888   698080         25% /boot  
tmpfs                401344       1408   399936          1% /run/user/1000  
[takonnova@fedora ~]$ du  
8      ./mozilla/extensions/{ec8030f7-c20a-464f-9b0e-13a3a9e97384}  
8      ./mozilla/extensions  
0      ./mozilla/plugins  
0      ./mozilla/firefox/Crash Reports/events  
0      ./mozilla/firefox/Crash Reports/pending  
12     ./mozilla/firefox/Crash Reports  
0      ./mozilla/firefox/Pending Pings  
0      ./mozilla/firefox/tzglg7me.default-release/minidumps  
0      ./mozilla/firefox/tzglg7me.default-release/crashes/events  
12     ./mozilla/firefox/tzglg7me.default-release/crashes  
2684   ./mozilla/firefox/tzglg7me.default-release/security_state  
0      ./mozilla/firefox/tzglg7me.default-release/storage/persistent/chrome/idb
```

Рис. 3.13: Выполнение команд df и du

Воспользовавшись справкой команды find (рис. [3.14]), выведем имена всех директорий, имеющих в домашнем каталоге. (рис. [3.15]).

```
takonnova@fedora:~ — 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. 3.14: man find


```
[takonnova@fedora ~]$ find ~/ -maxdepth 1 -type d -print
/home/takonnova/
/home/takonnova/.mozilla
/home/takonnova/.cache
/home/takonnova/.config
/home/takonnova/.local
/home/takonnova/Рабочий стол
/home/takonnova/Загрузки
/home/takonnova/Шаблоны
/home/takonnova/Общедоступные
/home/takonnova/Документы
/home/takonnova/Музыка
/home/takonnova/Изображения
/home/takonnova/Видео
/home/takonnova/.ssh
/home/takonnova/work
/home/takonnova/pandoc-2.19.2
/home/takonnova/.texlive2022
/home/takonnova/Архитектура
/home/takonnova/bin
```

Рис. 3.15: Вывод имён всех директорий в домашнем каталоге

4 Контрольные вопросы

1. Какие потоки ввода вывода вы знаете?

Ввод и вывод распределяется между тремя стандартными потоками: - `stdin` — стандартный ввод (клавиатура), - `stdout` — стандартный вывод (экран), - `stderr` — стандартная ошибка (вывод ошибок на экран).

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

Основное отличие: `>` : Перезаписывает существующий файл или создает файл, если файл с указанным именем отсутствует в каталоге. `»` : добавляет существующий файл или создает файл, если файл с указанным именем отсутствует в каталоге.

3. Что такое конвейер?

Конвейер (англ. `pipeline`) 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.

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

Процесс - это: - программа на стадии выполнения - “объект”, которому выделено процессорное время - асинхронная работа

5. Что такое PID и GID?

Идентификатор процесса (PID). Каждому новому процессу ядро присваивает уникальный идентификационный номер. В любой момент времени идентификатор процесса является уникальным, хотя после завершения процесса он может использоваться снова для другого процесса. Некоторые идентификаторы зарезервированы системой для особых процессов. Так, процесс с идентификатором 1 - это процесс инициализации `init`, являющийся предком всех других процессов в системе.

Идентификатор группы GID и эффективный идентификатор группы (EGID) GID - это идентификационный номер группы данного процесса. EGID связан с GID также, как EUID с UID.

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

Принудительное завершение процесса и изменение его приоритета) можно выполнить и без команды `top`. Процессы в Linux имеют возможность обмениваться так называемыми “сигналами” с ядром и другими процессами. При получении сигнала процессом, управление передается подпрограмме его обработки или ядру, если такой подпрограммы не существует. В Linux имеется команда `kill`, которая позволяет послать заданному процессу любой сигнал.

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

`top` - интерактивный просмотрщик процессов. `htop` аналог `top`. Программа `top` динамически выводит в режиме реального времени информации о работающей системе, т.е. о фактической активности процессов. По умолчанию она выдает задачи, наиболее загружающие процессор сервера, и обновляет список каждые две секунды.

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

`find` : Для поиска файлов из командной строки вы можете использовать команду “`find`”. У этой команды следующий синтаксис:

`find path criteria action` - “`path`” - Секция для указания директории поиска. Если ничего не указано поиск идет по текущей директории. - “`criteria`” - Опции поиска. - “`action`” - Опции, которые влияют на состояние поиска или контролируют его, например, - “`-print`”

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

Для поиска файла по содержимому проще всего воспользоваться командой `grep` (вместо `find`).

Пример: `grep -r строка_поиска каталог`

10. Как определить объем свободной памяти на жёстком диске?

Самый простой способ найти свободное место на диске в Linux - это используйте команду `df`. Команда `df` означает «свободное от диска» и, очевидно, показывает вам свободное и доступное дисковое пространство в системах Linux. Работы С Нами -`h` вариант, он показывает дисковое пространство в удобочитаемом формате (МБ и ГБ).

11. Как определить объем вашего домашнего каталога?

В операционных системах на базе Linux посмотреть размер папки (директории) можно с помощью команды `du`. Эта команда, выполняемая в консоли, позволяет оценить используемый объем места на жестком диске отдельно по папкам и файлам, просуммировать результат, узнать общий размер папки.

12. Как удалить зависший процесс?

Убиваем процессы в Linux — команды `ps`, `kill` и `killall`

- Находим PID зависшего процесса Каждый процесс в Linux имеет свой идентификатор, называемый PID.

- «Убиваем» процесс командой kill. Когда известен PID процесса, мы можем убить его командой kill.
- Убиваем процессы командой killall.
- Заключение

5 Выводы

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