

# **Отчет по выполнению лабораторной работы №8**

**Операционные системы**

Осина Виктория Александровна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
3.1	Перенаправление ввода-вывода . . . . .	7
3.2	Конвейер . . . . .	7
3.3	Поиск файла . . . . .	8
3.4	Фильтрация текста . . . . .	8
3.5	Проверка использования диска . . . . .	8
3.6	Управление задачами . . . . .	9
3.7	Управление процессами . . . . .	9
3.8	Получение информации о процессах . . . . .	9
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>10</b>
<b>5</b>	<b>Выводы</b>	<b>18</b>
	<b>Список литературы</b>	<b>19</b>

## Список иллюстраций

4.1	Запись названий файлов в новый файл . . . . .	10
4.2	Запись названий файлов в уже созданный файл . . . . .	10
4.3	Вывод содержимого файла file.txt . . . . .	11
4.4	Запись имен в новый файл и проверка . . . . .	11
4.5	Вывод файлов с именами, начинающимися на с . . . . .	11
4.6	Еще один вариант вывода . . . . .	12
4.7	Вывод имен файлов, начинающихся с символа h . . . . .	12
4.8	Запуск процесса в фоновом режиме . . . . .	12
4.9	Удаление файла . . . . .	12
4.10	Запуск редактора в фоновом режиме . . . . .	13
4.11	Определение идентификатора процесса . . . . .	13
4.12	Определение идентификатора процесса . . . . .	13
4.13	Название рисунка . . . . .	13
4.14	Название рисунка . . . . .	14
4.15	Завершение процесса при помощи kill . . . . .	14
4.16	Процесс завершен . . . . .	14
4.17	Чтение справки . . . . .	15
4.18	Чтение справки . . . . .	15
4.19	Выполнение команды df . . . . .	16
4.20	Выполнение команды du . . . . .	16
4.21	Чтение справки команды find . . . . .	16
4.22	Вывод имен директорий . . . . .	17

## **Список таблиц**

# 1 Цель работы

Целью данной лабораторной работы является ознакомление с инструментами поиска файлов и фильтрации текстовых данных, а также 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, имеющихся в вашем домашнем каталоге.

## 3 Теоретическое введение

### 3.1 Перенаправление ввода-вывода

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`. Например, команда `ls` выводит в стандартный поток вывода (консоль) список файлов в текущей директории. Потоки вывода и ввода можно перенаправлять на другие файлы или устройства. Проще всего это делается с помощью символов `>`, `>>`, `<`, `<<`.

### 3.2 Конвейер

Конвейер (`pipe`) 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

### 3.3 Поиск файла

Команда `find` используется для поиска и отображения на экран имён файлов, соответствующих заданной строке символов. Формат команды:

```
find путь [-опции]
```

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

### 3.4 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`.

### 3.5 Проверка использования диска

Команда `df` показывает размер каждого смонтированного раздела диска. Формат команды:

```
df [-опции] [файловая_система]
```

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

```
du [-опции] [имя_файла...]
```

На `afs` можно посмотреть использованное пространство командой

```
fs quota
```



## 3.6 Управление задачами

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 текстовый 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
```

## 3.7 Управление процессами

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.

## 3.8 Получение информации о процессах

Команда ps используется для получения информации о процессах. Формат команды:

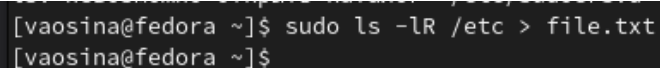
```
ps [-опции]
```

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

```
ps aux
```

## 4 Выполнение лабораторной работы

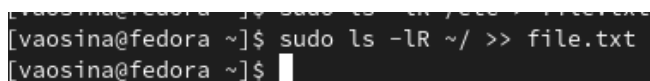
Вход в систему у меня уже был осуществлен, поэтому сразу приступаю к следующему заданию. Записываю в файл file.txt названия файлов, содержащихся в каталоге /etc, при помощи перенаправления “>”, чтобы создать файл file.txt. (рис. 4.1).



```
[vaosina@fedora ~]$ sudo ls -lR /etc > file.txt
[vaosina@fedora ~]$
```

Рис. 4.1: Запись названий файлов в новый файл

Затем дописываю в этот же файл названия файлов, содержащихся в моём домашнем каталоге, используя перенаправление “>>”. (рис. [4.2]).



```
[vaosina@fedora ~]$ sudo ls -lR ~/ >> file.txt
[vaosina@fedora ~]$
```

Рис. 4.2: Запись названий файлов в уже созданный файл

Вывожу имена всех файлов из file.txt, имеющих расширение .conf, (рис. [4.3]).

```
[vaosina@fedora ~]$ grep .conf file.txt
-rw-r--r--. 1 root root      269 янв 18  2023 anthy-unicode.conf
-rw-r--r--. 1 root root      833 фев 10  2023 appstream.conf
-rw-r--r--. 1 root root       55 янв 18  2023 asound.conf
-rw-r--r--. 1 root root    29842 янв 24  2023 brltty.conf
-rw-r--r--. 1 root root     1372 авг 29  2022 chrony.conf
drwxr-xr-x. 1 root root       18 апр 14  2023 dconf
-rw-r--r--. 1 root root     1280 янв 19  2023 dleyna-renderer-service.conf
-rw-r--r--. 1 root root     1174 янв 19  2023 dleyna-server-service.conf
-rw-r--r--. 1 root root    28485 фев 13  2023 dnsmasq.conf
-rw-r--r--. 1 root root       117 мар 14  2023 dracut.conf
drwxr-xr-x. 1 root root        0 мар 14  2023 dracut.conf.d
```

Рис. 4.3: Вывод содержимого файла file.txt

Теперь записываю их в новый текстовый файл conf.txt и при помощи команды head вывожу первые 10 имен для проверки.(рис. [4.4]).

```
[vaosina@fedora ~]$ grep .conf file.txt > conf.txt
[vaosina@fedora ~]$ head conf.txt
-rw-r--r--. 1 root root      269 янв 18  2023 anthy-unicode.conf
-rw-r--r--. 1 root root      833 фев 10  2023 appstream.conf
-rw-r--r--. 1 root root       55 янв 18  2023 asound.conf
-rw-r--r--. 1 root root    29842 янв 24  2023 brltty.conf
-rw-r--r--. 1 root root     1372 авг 29  2022 chrony.conf
drwxr-xr-x. 1 root root       18 апр 14  2023 dconf
-rw-r--r--. 1 root root     1280 янв 19  2023 dleyna-renderer-service.conf
-rw-r--r--. 1 root root     1174 янв 19  2023 dleyna-server-service.conf
-rw-r--r--. 1 root root    28485 фев 13  2023 dnsmasq.conf
-rw-r--r--. 1 root root       117 мар 14  2023 dracut.conf
```

Рис. 4.4: Запись имен в новый файл и проверка

Определяю, какие файлы в моём домашнем каталоге имеют имена, начинающиеся с символа с. (рис. [4.5]).

```
[vaosina@fedora ~]$ find ~ -name "с*" -print
/home/vaosina/.mozilla/firefox/9sth979l.default-release/compatibility.ini
/home/vaosina/.mozilla/firefox/9sth979l.default-release/cookies.sqlite
/home/vaosina/.mozilla/firefox/9sth979l.default-release/cert9.db
/home/vaosina/.mozilla/firefox/9sth979l.default-release/storage/permanent/chrome
/home/vaosina/.mozilla/firefox/9sth979l.default-release/storage/default/https+++mail.google.com/cache
```

Рис. 4.5: Вывод файлов с именами, начинающимися на с

Ещё один вариант, с помощью которого можно определить имена файлов, начинающихся на с. (рис. [4.6]).

```

/home/vaosina/conf.txt
[vaosina@fedora ~]$ ls -lR | grep c*
-rw-r--r--. 1 vaosina vaosina 48512 map 27 20:44 conf.txt
[vaosina@fedora ~]$

```

Рис. 4.6: Еще один вариант вывода

Вывожу на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h. (рис. [4.7]).

```

[vaosina@fedora ~]$ sudo find /etc -name "h*" -print
[sudo] пароль для vaosina:
/etc/avahi/hosts
/etc/brltty/Contraction/ha.ctb
/etc/brltty/Input/bm/horizontal.kti
/etc/brltty/Input/hd
/etc/brltty/Input/hm
/etc/brltty/Input/ht
/etc/brltty/Input/hw
/etc/brltty/Text/he.ttb

```

Рис. 4.7: Вывод имен файлов, начинающихся с символа h

Запускаю в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log. (рис. [4.8]).

```

[vaosina@fedora ~]$ find ~ -name "log*" > logfile &
[1] 54880
[vaosina@fedora ~]$

```

Рис. 4.8: Запуск процесса в фоновом режиме

Удаляю файл ~/logfile. (рис. [4.9]).

```

[vaosina@fedora ~]$ rm logfile
[1]+  Завершён      find ~ -name "log*" > logfile
[vaosina@fedora ~]$ ls
bin      file.txt  Видео    Загрузки  Музыка    'Рабочий стол'
conf.txt work     Документы  Изображения  Общедоступные  Шаблоны
[vaosina@fedora ~]$

```

Рис. 4.9: Удаление файла

Запускаю из консоли в фоновом режиме редактор gedit. (рис. [4.10]).

```
[vaosina@fedora ~]$ gedit &  
[vaosina@fedora ~]$
```

Рис. 4.10: Запуск редактора в фоновом режиме

Определяю идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep. (рис. [4.11]).

```
[vaosina@fedora ~]$ ps aux | grep gedit  
vaosina  54937  0.7  0.9 935012 78244 pts/0    Sl   20:52   0:01 gedit  
vaosina  54973  0.0  0.0 222436  2432 pts/0    S+   20:55   0:00 grep --color=auto gedit  
[vaosina@fedora ~]$
```

Рис. 4.11: Определение идентификатора процесса

Ещё один способ, при помощи которого можно определить идентификатор. (рис. [4.12]).

```
[vaosina@fedora ~]$ pgrep gedit  
54937  
[vaosina@fedora ~]$
```

Рис. 4.12: Определение идентификатора процесса

Читаю справку (man) команды kill (рис. [4.13]) и (рис. [4.14]).

```
[vaosina@fedora ~]$ man kill
```

Рис. 4.13: Название рисунка

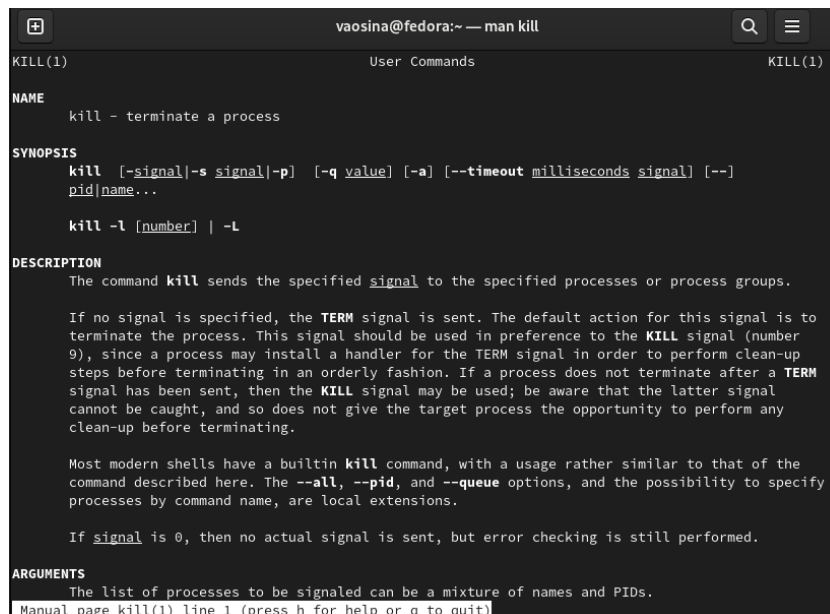
A screenshot of a terminal window showing the man page for the 'kill' command. The window title is 'vaosina@fedora:~ — man kill'. The content includes sections for NAME, SYNOPSIS, DESCRIPTION, and ARGUMENTS. The SYNOPSIS section shows the command syntax: 'kill [-signal|-s signal|-p] [-q value] [-a] [--timeout milliseconds signal] [--] pid|name...'. The DESCRIPTION section explains that the command sends a signal to specified processes or process groups, with the default being the TERM signal (number 9). The ARGUMENTS section states that the list of processes can be a mixture of names and PIDs. At the bottom, it says 'Manual page kill(1) line 1 (press h for help or q to quit)'.

Рис. 4.14: Название рисунка

Использую команду kill для завершения процесса gedit. (рис. [4.15]) и (рис. [4.16]).

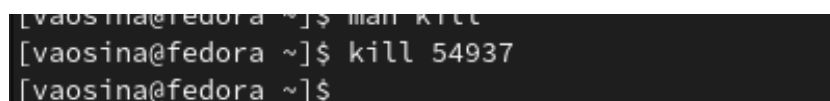
A screenshot of a terminal window showing the execution of the 'kill' command. The prompt is '[vaosina@fedora ~]\$'. The user enters 'man kill', then 'kill 54937', and finally '\$' to exit the command.

Рис. 4.15: Завершение процесса при помощи kill

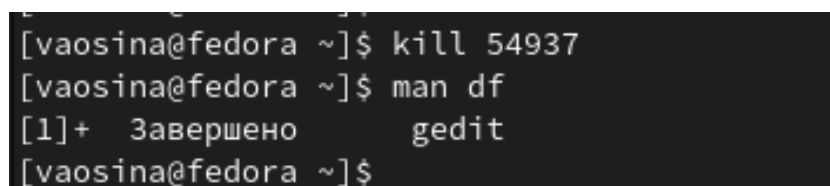
A screenshot of a terminal window showing the execution of the 'kill' command and the result of the 'df' command. The prompt is '[vaosina@fedora ~]\$'. The user enters 'kill 54937', then 'man df', and finally '\$'. The output of 'man df' shows '[1]+ Завершено gedit'.

Рис. 4.16: Процесс завершен

Читаю справку команды df. (рис. [4.17]).

```
vaosina@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 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

Manual page df(1) line 1/112 31% (press h for help or q to quit)
```

Рис. 4.17: Чтение справки

Читаю справку команды du. (рис. [4.18]).

```
vaosina@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 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, indirect 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

Manual page du(1) line 1/138 20% (press h for help or q to quit)
```

Рис. 4.18: Чтение справки

Выполняю команду df. (рис. [4.19]).

```
[vaosina@fedora ~]$ df -vi
Файловая система    Инодов    ИИспользовано    ИСвободно    ИИспользовано%    Смонтировано в
devtmpfs             1009588      508      1009080           1%    /dev
tmpfs                1014716       17      1014699           1%    /dev/shm
tmpfs                 819200      1152      818048           1%    /run
/dev/sda3              0           0           0             -      /
tmpfs                1048576       40      1048536           1%    /tmp
/dev/sda2              65536       387       65149           1%    /boot
/dev/sda3              0           0           0             -    /home
tmpfs                 202943       174      202769           1%    /run/user/1000
/dev/loop0             658         658           0          100%    /var/lib/napd/snap/snapd/20671
/dev/loop1            12041       12041           0          100%    /var/lib/napd/snap/core20/2182
/dev/loop3             6425        6425           0          100%    /var/lib/napd/snap/hugo/18786
/dev/loop4             651         651           0          100%    /var/lib/napd/snap/snapd/21184
/dev/loop5            14443       14443           0          100%    /var/lib/napd/snap/core22/1122
/dev/loop6             7633        7633           0          100%    /var/lib/napd/snap/hugo/18951
[vaosina@fedora ~]$
```

Рис. 4.19: Выполнение команды df

Выполняю команду du. (рис. [4.20]).

```
[vaosina@fedora ~]$ du -a work/
0      work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/branches
0      work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/hooks/applypatch-msg.sample
4      work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/hooks/commit-msg.sample
8      work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/hooks/fsmonitor-watchman.sample
4      work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/hooks/post-update.sample
4      work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/hooks/pre-applypatch.sample
4      work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/hooks/pre-commit.sample
4      work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/hooks/pre-merge-commit.sample
4      work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/hooks/pre-push.sample
8      work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/hooks/pre-rebase.sample
4      work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/hooks/pre-receive.sample
4      work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/hooks/prepare-commit-msg.sample
4      work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/hooks/push-to-checkout.sample
60     work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/hooks/update.sample
4      work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/hooks
4      work/study/2023-2024/Архитектура  компьютер/арх-пс/.git/info/exclude
```

Рис. 4.20: Выполнение команды du

Читаю справку команды find. (рис. [4.21]).

```
vaosina@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...] [expression]

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.
```

Рис. 4.21: Чтение справки команды find

Введу имена всех директорий, имеющиххся в моём домашнем каталоге. (рис. [4.22]).



```
[vaosina@fedora ~]$ find -type d
./
./mozilla
./mozilla/extensions
./mozilla/extensions/{ec8030f7-c20a-464f-9b0e-13a3a9e97384}
./mozilla/plugins
./mozilla/firefox
./mozilla/firefox/9sth979l.default-release
./mozilla/firefox/9sth979l.default-release/security_state
./mozilla/firefox/9sth979l.default-release/storage
./mozilla/firefox/9sth979l.default-release/storage/permanent
./mozilla/firefox/9sth979l.default-release/storage/permanent/chrome
./mozilla/firefox/9sth979l.default-release/storage/permanent/chrome/idb
```

Рис. 4.22: Вывод имен директорий

## 5 Выводы

Благодаря 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.

## **Список литературы**