

# Отчёт по лабораторной работе №6

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

Митичкина Екатерина Павловна

### Оглавление

<b>Цель работы.....</b>	<b>2</b>
<b>Задача.....</b>	<b>2</b>
<b>Теоретическое введение:.....</b>	<b>3</b>
Перенаправление ввода-вывода.....	3
Конвейер.....	3
Поиск файла.....	4
Фильтрация текста.....	4
Управление задачами.....	5
Управление процессами.....	6
Получение информации о процессах.....	6
<b>Выполнение лабораторной работы.....</b>	<b>6</b>
1. Осуществите вход в систему, используя соответствующее имя пользователя.....	6
2. Запишите в файл file.txt названия файлов, содержащихся в каталоге /etc. Допишите в этот же файл названия файлов, содержащихся в вашем домашнем каталоге.....	7
3. Выведите имена всех файлов из file.txt, имеющих расширение .conf, после чего запишите их в новый текстовый файл conf.txt.....	8
4. Определите, какие файлы в вашем домашнем каталоге имеют имена, начинавшиеся с символа с? Предложите несколько вариантов, как это сделать.....	9
5. Выведите на экран (по странично) имена файлов из каталога /etc, начинающиеся с символа h.....	9
6. Запустите в фоновом режиме процесс, который будет записывать в файл ~/logfile файлы, имена которых начинаются с log.....	10
7. Удалите файл ~/logfile.....	10
8. Запустите из консоли в фоновом режиме редактор gedit.....	11
9. Определите идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep. Как ещё можно определить идентификатор процесса?.....	11
10. Прочтите справку (man) команды kill, после чего используйте её для завершения процесса gedit.....	12

11. Выполните команды <code>df</code> и <code>du</code> , предварительно получив более подробную информацию об этих командах, с помощью команды <code>man</code> .	13
12. Воспользовавшись справкой команды <code>find</code> , выведите имена всех директорий, имеющих в вашем домашнем каталоге.	15
<b>Выводы</b>	<b>15</b>
<b>Ответы на контрольные вопросы</b>	<b>15</b>

## 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.

## Задача

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, начинавшиеся 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.fi`
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.

# Теоретическое введение:

## 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 >, >>, <, <<. Рассмотрим пример.

```
# Перенаправление stdout (вывода) в файл.  
# Если файл отсутствовал, то он создаётся, # иначе -- перезаписывается.  
# 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".
```

## 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  
# означает, что вывод команды 1 передаётся на ввод команде 2
```

Конвейеры можно группировать в цепочки и выводить с помощью перенаправления в файл, например:

```
ls -la |sort > sortilg_list
```

вывод команды **ls -la** передаётся команде сортировки **\*\*sort\*sorting\_list\verb**. **Чаше всего скрипты на Bash используются в качестве автоматизации каких-то рутинных**

операций в консоли, отсюда иногда возникает необходимость в обработке **stdout** одной команды и передача на **stdin\*\*** другой команде, при этом результат выполнения команды должен обработан.

## 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.

### Примеры:

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

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

Здесь **~** — 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
```

3. Найти в Вашем домашнем каталоге файлы, имена которых заканчиваются символом **~** и удалить их:

```
find ~ -name "*~" -exec rm "{}" \;
```

Здесь опция **-exec rm "{}"**; задаёт применение команды **rm** ко всем файлам, имена которых соответствуют указанной после опции **-name** строке символов.

Для просмотра опций команды **find** воспользуйтесь командой **man**.

## 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, содержащих в имени «лаб»: `` ls -l | grep лаб

## Проверка использования диска

Команда **df** показывает размер каждого смонтированного раздела диска.

Формат команды:

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

**Пример:**

```
df -vi
```

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
```

## 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 фоновыми программами 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
```

## 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.

## 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 вашем терминале, используйте опцию `ix`.

**Пример:**

`ps aux`

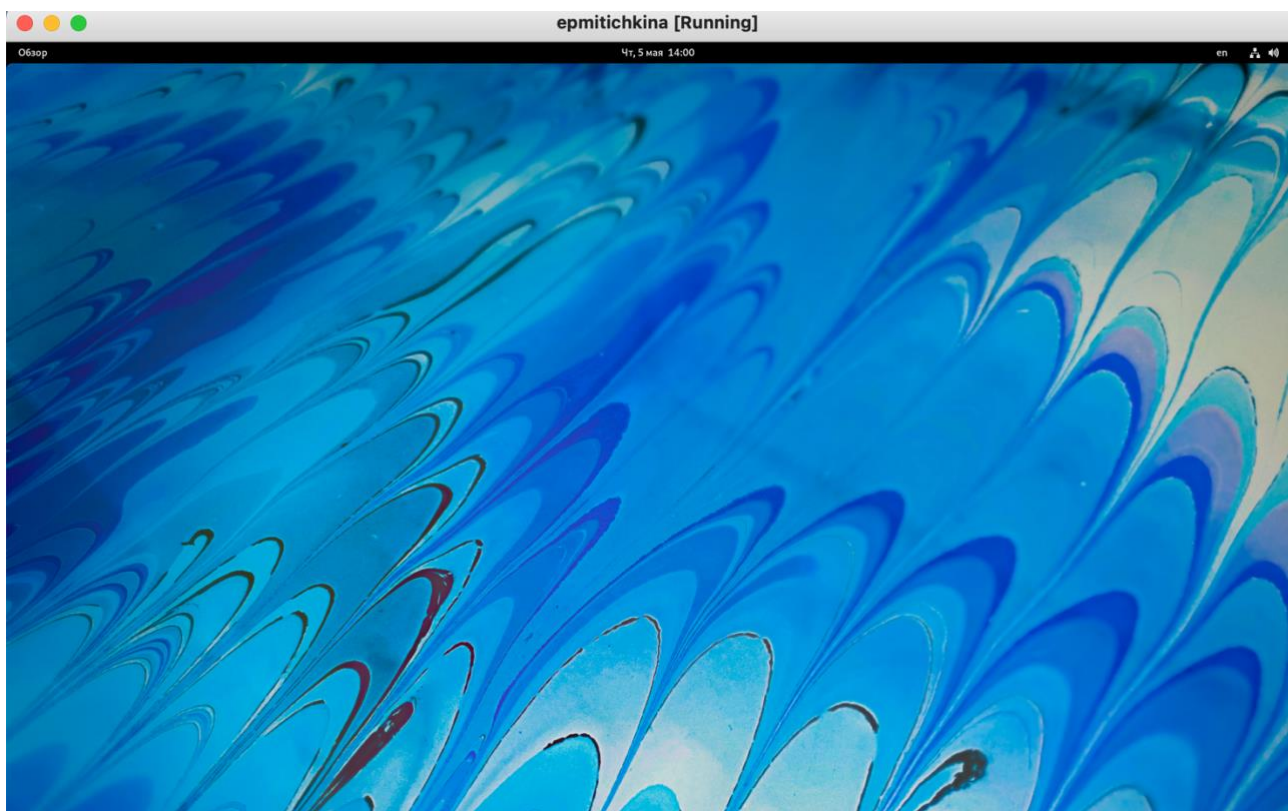
Для запуска команды в фоновом режиме необходимо в конце командной строки указать знак **&** (амперсанд).

Пример работы, требующей много машинного времени для выполнения, и которую целесообразно запустить в фоновом режиме:

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

## Выполнение лабораторной работы

1. Осуществите вход в систему, используя соответствующее имя пользователя.



вход в систему

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

```
[epmitichkina@fedora ~]$ find /etc -maxdepth 1 -type f > file.txt
[epmitichkina@fedora ~]$ find ~ -maxdepth 1 -type f >> file.txt
[epmitichkina@fedora ~]$ cat file.txt
/etc/.pwd.lock
/etc/.updated
/etc/DIR_COLORS
/etc/DIR_COLORS.lightbgcolor
/etc/GREP_COLORS
/etc/Trolltech.conf
/etc/adjtime
/etc/aliases
/etc/anthy-unicode.conf
/etc/appstream.conf
/etc/asound.conf
/etc/bashrc
/etc/bindresvport.blacklist
/etc/brlapi.key
/etc/brltty.conf
/etc/chrony.conf
/etc/chrony.keys
/etc/csh.cshrc
/etc/csh.login
```

запись файлы `/etc` и домашнего каталога

```
/etc/crypttab  
/etc/machine-id  
/etc/hostname  
/etc/group  
/etc/gshadow  
/etc/subuid-  
/etc/subuid  
/etc/subgid-  
/etc/subgid  
/etc/passwd  
/etc/shadow  
/etc/sensors3.conf  
/etc/ld.so.cache  
/home/epmitichkina/.bash_logout  
/home/epmitichkina/.bash_profile  
/home/epmitichkina/.bashrc  
/home/epmitichkina/.wget-hsts  
/home/epmitichkina/.vboxclient-clipboard.pid  
/home/epmitichkina/.vboxclient-seamless.pid  
/home/epmitichkina/.vboxclient-draganddrop.pid  
/home/epmitichkina/.gitconfig  
/home/epmitichkina/5  
/home/epmitichkina/file.txt
```

проверка

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

```
/etc/sensors3.conf  
[epmitichkina@fedora ~]$ grep '\.conf$' file.txt  
/etc/Trolltech.conf  
/etc/anthy-unicode.conf  
/etc/appstream.conf  
/etc/asound.conf  
/etc/brltty.conf  
/etc/chrony.conf  
/etc/dley-na-renderer-service.conf  
/etc/dley-na-server-service.conf  
/etc/dnsmasq.conf  
/etc/dracut.conf  
/etc/fprintd.conf  
/etc/fuse.conf  
/etc/host.conf  
/etc/idmapd.conf  
/etc/jwhois.conf  
/etc/kdump.conf  
/etc/krb5.conf  
/etc/ld.so.conf  
/etc/libaudit.conf  
/etc/libuser.conf  
/etc/locale.conf
```

поиск файло с расширение .conf



```
[epmitichkina@fedora ~]$ grep '\.conf$' file.txt >> conf.txt
[epmitichkina@fedora ~]$ cat conf.txt
/etc/Trolltech.conf
/etc/anthy-unicode.conf
/etc/appstream.conf
/etc/asound.conf
/etc/brltty.conf
/etc/chrony.conf
/etc/dleyna-renderer-service.conf
/etc/dleyna-server-service.conf
/etc/dnsmasq.conf
/etc/dracut.conf
/etc/fprintd.conf
/etc/fuse.conf
/etc/host.conf
/etc/idmapd.conf
/etc/jwhois.conf
/etc/kdump.conf
/etc/krb5.conf
/etc/ld.so.conf
/etc/libaudit.conf
/etc/libuser.conf
```

запись в conf.txt

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

```
[epmitichkina@fedora ~]$ find ~ -name c*
/home/epmitichkina/conf.txt
[epmitichkina@fedora ~]$ ls c*
conf.txt
[epmitichkina@fedora ~]$
```

файлы которые начинавшиеся с символа с

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

```
/etc/mercurial/hgr.conf
[epmitichkina@fedora ~]$ find /etc -maxdepth 1 -type f -name 'h*'
/etc/host.conf
/etc/hosts
/etc/hostname
[epmitichkina@fedora ~]$
```

файлы которые начинавшиеся с символа h

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

```
[epmitichkina@fedora ~]$ find ~ -name 'log*' > logfile &
[1] 8390
[epmitichkina@fedora ~]$ cat logfile
/home/epmitichkina/.mozilla/firefox/dp2js3z5.default-release/logins-backup.json
/home/epmitichkina/.mozilla/firefox/dp2js3z5.default-release/logins.json
/home/epmitichkina/.local/share/keyrings/login.keyring
/home/epmitichkina/.local/share/Trash/files/blog/.git/logs
/home/epmitichkina/work/study/2022-2023/Операционные системы/study_2022-2023_os-intro/.git/logs
/home/epmitichkina/work/study/2022-2023/Операционные системы/study_2022-2023_os-intro/.git/modules/templa
/home/epmitichkina/work/study/2022-2023/Операционные системы/study_2022-2023_os-intro/.git/modules/templa
/home/epmitichkina/work/blog/.git/logs
/home/epmitichkina/work/blog/.git/modules/public/logs
/home/epmitichkina/work/setting/.git/logs
/home/epmitichkina/logfile
```

фоновый режим

7. Удалите файл ~/logfile

```
[epmitichkina@fedora ~]$ rm logfile
[epmitichkina@fedora ~]$ ls
01  conf.txt  montly  Видео  Загрузки  Музыка  'Рабочий стол'
5   file.txt  work    Документы  Изображения  Общедоступные  Шаблоны
[epmitichkina@fedora ~]$
```

удаление logfile

## 8. Запустите из консоли в фоновом режиме редактор gedit.

```
/home/epmitichkina/work/study/2022-2023/Операционные системы/study_2022-2023_os-intro/.git/modules/templa
Открыть ▾ + Без имени 1 Сохранить ≡ x
1
ins-backup.json
ins.json
dy_2022-2023_os-intro/.git/logs
dy_2022-2023_os-intro/.git/modules/templa
dy_2022-2023_os-intro/.git/modules/templa
Текст ▾ Ширина табуляций: 8 ▾ Стр 1, Стлб 1 ▾ ВСТ
[epmitichkina@fedora ~]$ gedit &
[1] 8440
[epmitichkina@fedora ~]$
```

запуск gedit

## 9. Определите идентификатор процесса gedit, используя команду ps, конвейер и фильтр grep. Как ещё можно определить идентификатор процесса?

```
[epmitichkina@fedora ~]$ gedit &
[1] 8440
[epmitichkina@fedora ~]$ ps
  PID TTY          TIME CMD
 8230 pts/1    00:00:00 bash
 8440 pts/1    00:00:00 gedit
 8478 pts/1    00:00:00 ps
[epmitichkina@fedora ~]$ ps | grep 'gedit'
 8440 pts/1    00:00:00 gedit
[epmitichkina@fedora ~]$
```

идентификатор процесса gedit

```

KILL(1)
User Commands

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. The signal should be used in preference to the KILL signal (number 9), since a process may install a handler in order to perform clean-up steps before terminating in an orderly fashion. If a process does not have a handler for a signal, then the KILL signal may be used; be aware that the latter signal cannot be caught or ignored.

    Most modern shells have a builtin kill command, with a usage rather similar to that of the command kill. The --pid, and --queue options, and the possibility to specify processes by command name, are local extensions.

    If signal is 0, then no actual signal is sent, but error checking is still performed.

ARGUMENTS
    The list of processes to be signaled can be a mixture of names and PIDs.

    pid
        Each pid can be expressed in one of the following ways:

        n
            where n is larger than 0. The process with PID n is signaled.

        0
            All processes in the current process group are signaled.

        -1
            All processes with a PID larger than 1 are signaled.

        -n
            where n is larger than 1. All processes in process group n are signaled. When an argument is given as a process name, it is taken literally.

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

[epmitichkina@fedora ~]$ ps -ef | grep gedit
epmitic+  8440    8230  0 13:43 pts/1    00:00:00 gedit
epmitic+  8579    8230  0 13:52 pts/1    00:00:00 grep --color=auto gedit
[epmitichkina@fedora ~]$ kill -KILL 8440
[1]+  Убито                  gedit
[epmitichkina@fedora ~]$
завершения процесса gedit

```

## 11. Выполните команды `df` и `du`, предварительно получив более подробную информацию об этих командах, с помощью команды `man`.

DF(1)

User Commands

### 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 each file name argument. If no file name is given, the space available on all currently mounted file systems is shown in 1K blocks by default, unless the environment variable `POSIXLY_CORRECT` is set, in which case blocks are used.

If an argument is the absolute file name of a disk device node containing a mounted file system, `df` displays the space available on that file system rather than on the file system containing the device node. This version of `df` is available on unmounted file systems, because on most kinds of systems doing so requires very nonportable 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

**--direct**

show statistics for a file instead of mount point

**-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)

спрака df

```
[epmitichkina@fedora ~]$ df
Файловая система 1K-блоков  Использовано  Доступно  Использовано%  Смонтировано в
devtmpfs           986252            0   986252            0% /dev
tmpfs              1006644            0   1006644            0% /dev/shm
tmpfs              402660           1368   401292            1% /run
/dev/sda2          82836480        5693252  75421964            8% /
tmpfs              1006644            68   1006576            1% /tmp
/dev/sda2          82836480        5693252  75421964            8% /home
/dev/sda1           996780         175308   752660           19% /boot
tmpfs              201328           136   201192            1% /run/user/1000
[epmitichkina@fedora ~]$
```

выполнение df

**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 apparent size is usually smaller than the disk usage, it includes 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

**-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; **--max-depth=0** is the same as **--summarize**

**--files0-from=F**

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

**спрака du**

```

[epmitichkina@fedora ~]$ du
8      ./mozilla/extensions/{ec8030f7-c20a-464f-9b0e-13a3a9e97384}
8      ./mozilla/extensions
0      ./mozilla/plugins
0      ./mozilla/firefox/Crash Reports/events
4      ./mozilla/firefox/Crash Reports
0      ./mozilla/firefox/Pending Pings
0      ./mozilla/firefox/dp2js3z5.default-release/minidumps
0      ./mozilla/firefox/dp2js3z5.default-release/crashes/events
4      ./mozilla/firefox/dp2js3z5.default-release/crashes
2692   ./mozilla/firefox/dp2js3z5.default-release/security_state
0      ./mozilla/firefox/dp2js3z5.default-release/storage/permanent/chrome/idb/3870112724rsegmnoittet-e
772    ./mozilla/firefox/dp2js3z5.default-release/storage/permanent/chrome/idb/3870112724rsegmnoittet-e
0      ./mozilla/firefox/dp2js3z5.default-release/storage/permanent/chrome/idb/3561288849sdhlie.files
0      ./mozilla/firefox/dp2js3z5.default-release/storage/permanent/chrome/idb/1451318868ntouromlalnodr
0      ./mozilla/firefox/dp2js3z5.default-release/storage/permanent/chrome/idb/1657114595AmcateirvtiSty

```

**выполнение du**



12. Воспользовавшись справкой команды `find`, выведите имена всех директорий, имеющих в вашем домашнем каталоге.

```
тсссиа .  
[epmitichkina@fedora ~]$ find ~ -maxdepth 1 -type d  
/home/epmitichkina  
/home/epmitichkina/.mozilla  
/home/epmitichkina/.cache  
/home/epmitichkina/.config  
/home/epmitichkina/.local  
/home/epmitichkina/Рабочий стол  
/home/epmitichkina/Загрузки  
/home/epmitichkina/Шаблоны  
/home/epmitichkina/Общедоступные  
/home/epmitichkina/Документы  
/home/epmitichkina/Музыка  
/home/epmitichkina/Изображения  
/home/epmitichkina/Видео  
/home/epmitichkina/.ssh  
/home/epmitichkina/.gnupg  
/home/epmitichkina/work  
/home/epmitichkina/01  
/home/epmitichkina/monthly  
[epmitichkina@fedora ~]$
```

вывод всех директорий домашнего каталога

## Выводы

В результате работы ознакомилась с инструментами поиска файлов и фильтрации текстовых данных. Приобрела практических навыков: по управлению процессами (и заданиями), по проверке использования диска и обслуживанию файловых систем.

## Ответы на контрольные вопросы

1. Какие потоки ввода вывода вы знаете?

- `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 >>.

Операция ">" перезаписывает файл вывода новыми данными, а ">>" осуществляет дописывание в конец указанного файла.

3. Что такое конвейер?

Конвейер - это оператор, объединяющий команды в цепочки, в которой результат работы предыдущей команды передается следующей.

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

Программа - это набор инструкций исполнителю, процесс - сущность исполняемой программы. Процесс содержит текст программы, стек ее данных, дескрипторы открытых файлов и другую служебную информацию.

5. Что такое PID и GID?

- PID - это идентификатор процесса, его уникальный номер.
- GID - это идентификатор группы пользователя в Linux.

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

Задачами называются запущенные фоновые программы. Узнать список запущенных в данный момент задач можно с помощью команды `jobs`, а посылать им управляющие сигналы можно с помощью команды `kill`.

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

- `top` (table of processes) - консольная команда, выводящая список работающих в системе процессов и информацию о них. По умолчанию `top` сортирует все процессы в реальном времени по их нагрузке на процессор.
- `htop` - Компьютерная программа, предназначенная для вывода на терминал списка запущенных процессов и информации о них. Создана как альтернатива программе `top`. Написана на языке Си. Предоставляет пользователю текстовый интерфейс, для вывода на терминал использует библиотеку `ncurses`.

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

Команда `find` используется для поиска и отображения на экран имен 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`

Здесь `~` — обозначение вашего домашнего каталога, `-name` — после этой опции указывается имя файла, который нужно найти, `"f*"` — строка символов, опреде



ляющая 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
```

3. Найти в Вашем домашнем каталоге файлы, имена которых заканчиваются символом `~` и удалить их:

```
find ~ -name "*~" -exec rm "{}" ; ``
```

Здесь опция `-exec rm "{}"`; задаёт применение команды `rm` ко всем файлам, имена которых соответствуют указанной после опции `-name` строке символов.

9. Можно ли по контексту (содержанию) найти файл? Если да, то как?  
С помощью команды `grep` можно искать файлы по содержимому. Для этого нужно вызвать команду `grep` с опциями `-lir` и искомым текстом, например `grep -lir '#include'`.

10. Как определить объем свободной памяти на жёстком диске?  
Набрав в консоли

```
df -h
```

можно увидеть объем свободной памяти в каждой файловой системе.  
Просуммировав эти значения можно узнать объем свободной памяти на жёстком диске.

11. Как определить объем вашего домашнего каталога?  
Объем домашнего каталога в удобочитаемом формате можно узнать набрав в терминале

```
du ~ -h
```

12. Как удалить зависший процесс?  
Удалить зависший процесс с идентификатором `pid` можно набрав в терминале `kill -KILL pid`.