

Отчёт по лабораторной работе №5

Анализ файловой системы Linux

Митичкина Екатерина Павловна

Оглавление

Цель работы.....	2
Задача	2
Теоретическое введение:.....	3
Формат команды.....	3
Команда man	3
Команда cd.	4
Команда pwd	4
Команда ls.....	5
Команда mkdir.....	6
Команда rm	7
Команды для работы с файлами и каталогами	8
touch.....	8
cat.....	8
less.....	8
head.....	8
tail.....	8
Копирование файлов и каталогов.....	8
cp	8
Перемещение и переименование файлов и каталогов.....	9
Права доступа	10
Изменение прав доступа	10
Анализ файловой системы.....	11
mount.....	12
df.....	13
Выполнение лабораторной работы	13
1. Выполните все примеры, приведённые в первой части описания лабораторной работы.	13
Пример №1	13
Пример №2	14
Пример №3	14
Пример №4	15
Пример №5	16
2. Выполните следующие действия, зафиксировав в отчёте по лабораторной работе используемые при этом команды и результаты их выполнения:	17

3. Определите опции команды <code>chmod</code> , необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет:	18
4. Прodelайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды:	19
5. Прочитайте <code>man</code> по командам <code>mount</code> , <code>fsck</code> , <code>mkfs</code> , <code>kill</code> и кратко их охарактеризуйте, приведя примеры.	21
Выводы	24
Ответы на контрольные вопросы	24

Цель работы

Ознакомление с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобретение практических навыков по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

Задача

1. Выполните все примеры, приведённые в первой части описания лабораторной работы.
2. Выполните следующие действия, зафиксировав в отчёте по лабораторной работе используемые при этом команды и результаты их выполнения:
 1. Скопируйте файл `/usr/include/sys/io.h` в домашний каталог и назовите его `equipment`. Если файла `io.h` нет, то используйте любой другой файл в каталоге `/usr/include/sys/` вместо него.
 2. В домашнем каталоге создайте директорию `~/ski.places`.
 3. Переместите файл `equipment` в каталог `~/ski.places`.
 4. Переименуйте файл `~/ski.places/equipment` в `~/ski.places/equiplist`.
 5. Создайте в домашнем каталоге файл `abc1` и скопируйте его в каталог `~/ski.places`, назовите его `equiplist2`.
 6. Создайте каталог с именем `equipment` в каталоге `~/ski.places`.
 7. Переместите файлы `~/ski.places/equiplist` и `equiplist2` в каталог `~/ski.places/equipment`.
 8. Создайте и переместите каталог `~/newdir` в каталог `~/ski.places` и назовите его `plans`.
3. Определите опции команды `chmod`, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет:
 1. `drwxr--r-- ... australia`
 2. `drwx--x--x ... play`

3. `-r-xr--r-- ... my_os`
4. `-rw-rw-r-- ... feathers` При необходимости создайте нужные файлы.
4. Прделайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды:
 1. Просмотрите содержимое файла `/etc/password`.
 2. Скопируйте файл `~/feathers` в файл `~/file.old`.
 3. Переместите файл `~/file.old` в каталог `~/play`.
 4. Скопируйте каталог `~/play` в каталог `~/fun`.
 5. Переместите каталог `~/fun` в каталог `~/play` и назовите его `games`.
 6. Лишите владельца файла `~/feathers` права на чтение.
 7. Что произойдёт, если вы попытаетесь просмотреть файл `~/feathers` командой `cat`?
 8. Что произойдёт, если вы попытаетесь скопировать файл `~/feathers`?
 9. Дайте владельцу файла `~/feathers` право на чтение.
 10. Лишите владельца каталога `~/play` права на выполнение.
 11. Перейдите в каталог `~/play`. Что произошло?
 12. Дайте владельцу каталога `~/play` право на выполнение.
5. Прочитайте `man` по командам `mount`, `fsck`, `mkfs`, `kill` и кратко их охарактеризуйте, приведя примеры.

Теоретическое введение:

Формат команды

Командой в операционной системе называется записанный по специальным правилам текст (возможно с аргументами), представляющий собой указание на выполнение какой-либо функций (или действий) в операционной системе. Обычно первым словом идёт имя команды, остальной текст — аргументы или опции, конкретизирующие действие.

Общий формат команд можно представить следующим образом:

`<имя_команды><разделитель><аргументы>`.

Команда `man`

Команда `man` используется для просмотра (оперативная помощь) в диалоговом режиме руководства (`manual`) по основным командам операционной системы типа Linux.

Формат команды: `man <команда>`

Пример (вывод информации о команде `man`): `man man`.

Для управления просмотром результата выполнения команды `man` можно использовать следующие клавиши: - Space — перемещение по документу на одну страницу вперёд; - Enter — перемещение по документу на одну строку вперёд; - q — выход из режима просмотра описания.

Команда cd.

Команда `cd` используется для перемещения по файловой системе операционной системы типа Linux.

Замечание 1. Файловая система ОС типа Linux — иерархическая система каталогов, подкаталогов и файлов, которые обычно организованы и сгруппированы по функциональному признаку.

Самый верхний каталог в иерархии называется корневым и обозначается символом `/`. Корневой каталог содержит системные файлы и другие каталоги.

Формат команды: `cd [путь_к_каталогу]`

Для перехода в домашний каталог пользователя следует использовать команду `cd` без параметров или `cd ~`.

Например, команда `cd /afs/dk.sci.pfu.edu.ru/home` позволяет перейти в каталог `/afs/dk.sci.pfu.edu.ru/home` (если такой существует), а для того, чтобы подняться выше на одну директорию, следует использовать: `cd ..`

Подробнее об опциях команды **cd** смотри в справке с помощью команды `man cd`.

Команда pwd

Для определения абсолютного пути к текущему каталогу используется команда `pwd` (print working directory).

Пример (абсолютное имя текущего каталога пользователя dharma):

```
pwd
результат:
1 /afs/dk.sci.pfu.edu.ru/home/d/h/dharma
```

Сокращения имён файлов. В работе с командами, в качестве аргументов которых выступает путь к какому-либо каталогу или файлу, можно использовать сокращённую запись пути. Символы сокращения приведены в табл. 4.1.

Таблица 4.1 - ~ Домашний каталог - . Текущий каталог - .. Родительский каталог

Например, в команде `cd` для перемещения по файловой системе сокращённую запись пути можно использовать следующим образом (команды чередуются с выводом результата выполнения команды `pwd`):

```
pwd

/afs/dk.sci.pfu.edu.ru/home/d/h/dharma

cd ..
pwd

/afs/dk.sci.pfu.edu.ru/home/d/h
```

```
cd ../../  
pwd
```

```
/afs/dk.sci.pfu.edu.ru/home
```

```
cd ~/work  
pwd
```

```
/afs/dk.sci.pfu.edu.ru/home/d/h/dharma/work
```

Команда ls

Команда ls используется для просмотра содержимого каталога.

Формат команды: ls [-опции] [путь]

Пример:

```
cd  
cd ..  
pwd
```

```
/afs/dk.sci.pfu.edu.ru/home/d/h
```

```
ls
```

```
dharma
```

Некоторые файлы в операционной системе скрыты от просмотра и обычно используются для настройки рабочей среды. Имена таких файлов начинаются с точки. Для того, чтобы отобразить имена скрытых файлов, необходимо использовать команду **ls** с опцией **a**: **ls -a**.

Можно также получить информацию о типах файлов (каталог, исполняемый файл, ссылка), для чего используется опция **F**. При использовании этой опции в поле имени выводится символ, который определяет тип файла (см. табл. 4.2) Таблица 4.2 - Каталог / - Исполняемый файл * - Ссылка @

Чтобы вывести на экран подробную информацию о файлах и каталогах, необходимо использовать опцию **l**. При этом о каждом файле и каталоге будет выведена следующая информация: - тип файла, - право доступа, - число ссылок, - владелец, - размер, - дата последней ревизии, - имя файла или каталога.

Пример:

```
cd /  
ls
```

Результат:

```
bin boot dev etc home lib media mnt  
opt proc root sbin sys tmp usr var
```

В этом же каталоге команда `ls -aF` даст примерно следующий результат:

```
drwxr-xr-x 21 root root 4096 Jan. 17 09:00 ./
drwxr-xr-x 21 root root 4096 Jan. 17 09:00 ../
drwxr-xr-x 2 root root 4096 Jan. 18 15:57 bin/
drwxr-xr-x 2 root root 4096 Apr. 14 2008 boot/
drwxr-xr-x 20 root root 14120 Feb. 17 10:48 dev/
drwxr-xr-x 170 root root 12288 Feb. 17 09:19 etc/
drwxr-xr-x 6 root root 4096 Aug. 5 2009 home/
lrwxrwxrwx 1 root root 5 Jan. 12 22:01 lib -> lib64/
drwxr-xr-x 8 root root 4096 Jan. 30 21:41 media/
drwxr-xr-x 5 root root 4096 Jan. 17 2010 mnt/
drwxr-xr-x 25 root root 4096 Jan. 16 09:55 opt/
dr-xr-xr-x 163 root root 0 Feb. 17 13:17 proc/
drwxr-xr-x 31 root root 4096 Feb. 15 23:57 root/
drwxr-xr-x 2 root root 12288 Jan. 18 15:57 sbin/
drwxr-xr-x 12 root root 0 Feb. 17 13:17 sys/
drwxrwxrwt 12 root root 500 Feb. 17 16:35 tmp/
drwxr-xr-x 22 root root 4096 Jan. 18 09:26 usr/
drwxr-xr-x 17 root root 4096 Jan. 14 17:38 var/
```

Команда `mkdir`

Команда `mkdir` используется для создания каталогов.

Формат команды: `mkdir имя_каталога1 [имя_каталога2...]`

Пример создания каталога в текущем каталоге:

```
cd
pwd
```

```
/afs/dk.sci.pfu.edu.ru/home/d/h/dharma
```

```
ls
```

```
Desktop public tmp
GNUstep public_html work
```

```
mkdir abc
```

```
ls
```

```
abc GNUstep public_html work
Desktop public tmp
```

Замечание 2. Для того чтобы создать каталог в определённом месте файловой системы, должны быть правильно установлены права доступа.

Можно создать также подкаталог в существующем подкаталоге:

```
mkdir parentdir
mkdir parentdir/dir
```

При задании нескольких аргументов создаётся несколько каталогов:

```
cd parentdir  
mkdir dir1 dir2 dir3
```

Можно использовать группировку: `mkdir parentdir/{dir1,dir2,dir3}`

Если же требуется создать подкаталог в каталоге, отличном от текущего, то путь к нему требуется указать в явном виде:

```
mkdir ../dir1/dir2  
или  
mkdir ~/dir1/dir2
```

Интересны следующие опции: - **--mode** (или **-m**) — установка атрибутов доступа; - **--parents** (или **-p**) — создание каталога вместе с родительскими по отношению к нему каталогами.

Атрибуты задаются в численной или символьной нотации: `mkdir --mode=777 dir` или `mkdir -m a+rwX dir`

Опция **--parents** (краткая форма **-p**) позволяет создавать иерархическую цепочку подкаталогов, создавая все промежуточные каталоги: `mkdir -p ~/dir1/dir2/dir3`

Команда `rm`

Команда `rm` используется для удаления файлов и/или каталогов.

Формат команды: `rm [-опции] [файл]`

Если требуется, чтобы выдавался запрос подтверждения на удаление файла, то необходимо использовать опцию **i**.

Чтобы удалить каталог, содержащий файлы, нужно использовать опцию **r**. Без указания этой опции команда не будет выполняться.

Пример:

```
cd  
mkdir abs  
rm abc
```

```
rm: abc is a directory
```

```
rm -r abc
```

Если каталог пуст, то можно воспользоваться командой **rmdir**. Если удаляемый каталог содержит файлы, то команда не будет выполнена — нужно использовать `rm -r имя_каталога`.

Команды для работы с файлами и каталогами

touch

Для создания текстового файла можно использовать команду **touch**. Формат команды: touch имя-файла

cat

Для просмотра файлов небольшого размера можно использовать команду **cat**.

Формат команды: cat имя-файла

less

Для просмотра файлов постранично удобнее использовать команду **less**.

Формат команды: less имя-файла

Следующие клавиши используются для управления процессом просмотра: - Space — переход к следующей странице, - ENTER — сдвиг вперёд на одну строку, - b — возврат на предыдущую страницу, - h — обращение за подсказкой, - q — выход из режима просмотра файла.

head

Команда **head** выводит по умолчанию первые 10 строк файла.

Формат команды: head [-n] имя-файла, где n — количество выводимых строк.

tail

Команда **tail** выводит по умолчанию 10 последних строк файла.

Формат команды: tail [-n] имя-файла, где n — количество выводимых строк.

Копирование файлов и каталогов

cp

Команда **cp** используется для копирования файлов и каталогов.

Формат команды: cp [-опции] исходный_файл целевой_файл

Примеры:

1. Копирование файла в текущем каталоге. Скопировать файл ~/abc1 в файл april и в файл may:

```
cd
touch abc1
cp abc1 april
cp abc1 may
```


2. Копирование нескольких файлов в каталог. Скопировать файлы april и may в каталог monthly:

```
mkdir monthly
cp april may monthly
```

3. Копирование файлов в произвольном каталоге. Скопировать файл monthly/may в файл с именем june:

```
cp monthly/may monthly/june
ls monthly
```

Опция **i** в команде **cp** выведет на экран запрос подтверждения о перезаписи файла. Для рекурсивного копирования каталогов, содержащих файлы, используется команда **cp** с опцией **r**.

Примеры:

1. Копирование каталогов в текущем каталоге. Скопировать каталог monthly в каталог monthly.00:

```
mkdir monthly.00
cp -r monthly monthly.00
```

2. Копирование каталогов в произвольном каталоге. Скопировать каталог monthly.00 в каталог /tmp

```
cp -r monthly.00 /tmp
```

Перемещение и переименование файлов и каталогов

Команды **mv** и **mkdir** предназначены для перемещения и переименования файлов и каталогов.

Формат команды **mv**: **mv** [-опции] старый_файл новый_файл

Примеры: 1. Переименование файлов в текущем каталоге. Изменить название файла april на july в домашнем каталоге:

```
cd
mv april july
```

2. Перемещение файлов в другой каталог. Переместить файл july в каталог monthly.00:

```
mv july monthly.00
ls monthly.00
```

Результат:

```
april july june may
```

Если необходим запрос подтверждения о перезаписи файла, то нужно использовать опцию **i**.

3. Переименование каталогов в текущем каталоге. Переименовать каталог `monthly.00` в `monthly.01`

```
mv monthly.00 monthly.01
```

4. Перемещение каталога в другой каталог. Переместить каталог `monthly.01` в каталог `reports`:

```
mkdir reports
mv monthly.01 reports
```

5. Переименование каталога, не являющегося текущим. Переименовать каталог `reports/monthly.01` в `reports/monthly`:

```
mv reports/monthly.01 reports/monthly
```

Права доступа

Каждый файл или каталог имеет права доступа. В сведениях о файле или каталоге указываются: - тип файла (символ (-) обозначает файл, а символ (d) — каталог); - права для владельца файла (r — разрешено чтение, w — разрешена запись, x — разрешено выполнение, - — право доступа отсутствует); - права для членов группы (r — разрешено чтение, w — разрешена запись, x — разрешено выполнение, - — право доступа отсутствует); - права для всех остальных (r — разрешено чтение, w — разрешена запись, x — разрешено выполнение, - — право доступа отсутствует).

Примеры:

1. Для файла (крайнее левое поле имеет значение -) владелец файла имеет право на чтение и запись (rw-), группа, в которую входит владелец файла, может читать файл (r--), все остальные могут читать файл (r--): -rw-r--r--
2. Только владелец файла имеет право на чтение, изменение и выполнение файла: -rwx-----.
3. Владелец каталога (крайнее левое поле имеет значение d) имеет право на просмотр, изменение и доступа в каталог, члены группы могут входить и просматривать его, все остальные — только входить в каталог: drwxr-x--x.

Изменение прав доступа

Права доступа к файлу или каталогу можно изменить, воспользовавшись командой **chmod**. Сделать это может владелец файла (или каталога) или пользователь с правами администратора.

Формат команды: `chmod режим имя_файла`

Режим (в формате команды) имеет следующие компоненты структуры и способ записи: - = установить право - - лишить права - + дать право - r чтение - w запись - u (user) владелец файла - g (group) группа, к которой принадлежит владелец файла - o (others) все остальные

В работе с правами доступа можно использовать их цифровую запись (восьмеричное значение) вместо символьной. Формы записи прав доступа

Двоичная	Восьмеричная	Символьная
111	7	rxw
110	6	rw
101	5	r-x
100	4	r--
011	3	-wx
010	2	-w
001	1	--x
000	0	---

Примеры:

1. Требуется создать файл ~/may с правом выполнения для владельца:

```
cd
touch may
ls -l may
chmod u+x may
ls -l may
```

2. Требуется лишить владельца файла ~/may права на выполнение:

```
chmod u-x may
ls -l may
```

3. Требуется создать каталог monthly с запретом на чтение для членов группы и всех остальных пользователей:

```
cd
mkdir monthly
chmod g-r, o-r monthly
```

4. Требуется создать файл ~/abc1 с правом записи для членов группы:

```
1 cd
2 touch abc1
3 chmod g+w abc1
```

Анализ файловой системы

Файловая система в Linux состоит из файлов и каталогов. Каждому физическому носителю соответствует своя файловая система.

Существует несколько типов файловых систем. Перечислим наиболее часто встречающиеся типы: - ext2fs (second extended filesystem); - ext3fs (third extended file system); - ext4 (fourth extended file system); - ReiserFS; - xfs; - fat (file allocation table); - ntfs (new technology file system).

mount

Для просмотра используемых в операционной системе файловых систем можно воспользоваться командой **mount** без параметров. В результате её применения можно получить примерно следующее:

```
mount
```

```
proc on /proc type proc (rw)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec)
udev on /dev type tmpfs (rw,nosuid)
devpts on /dev/pts type devpts (rw,nosuid,noexec)
/dev/sda1 on /mnt/a type ext3 (rw,noatime)
/dev/sdb2 on /mnt/docs type reiserfs (rw,noatime)
shm on /dev/shm type tmpfs (rw,noexec,nosuid,nodev)
usbfs on /proc/bus/usb type usbfs
(rw,noexec,nosuid,devmode=0664,devgid=85)
binfmt_misc on /proc/sys/fs/binfmt_misc type binfmt_misc
(rw,noexec,nosuid,nodev)
nfsd on /proc/fs/nfs type nfsd (rw,noexec,nosuid,nodev)
```

В данном случае указаны имена устройств, названия соответствующих им точек монтирования (путь), тип файловой системы и параметрами монтирования.

В контексте команды mount устройство — специальный файл устройства, с помощью которого операционная система получает доступ к аппаратному устройству.

Файлы устройств обычно располагаются в каталоге /dev, имеют сокращённые имена (например, sdaN, sdbN или hdaN, hdbN, где N — порядковый номер устройства, sd — устройства SCSI, hd — устройства MFM/IDE).

Точка монтирования — каталог (путь к каталогу), к которому присоединяются файлы устройств.

Другой способ определения смонтированных в операционной системе файловых систем — просмотр файла /etc/fstab. Сделать это можно например с помощью команды

```
cat /etc/fstab
```

```
/dev/hda1 / ext2 defaults 1 1
/dev/hda5 /home ext2 defaults 1 2
/dev/hda6 swap swap defaults 0 0
/dev/hdc /mnt/cdrom auto umask=0,user,noauto,ro,exec,users 0 0
```

```
none /mnt/floppy supermount dev=/dev/fd0,fs=ext2:vfat,--,
sync,umask=0 0 0
none /proc proc defaults 0 0
none /dev/pts devpts mode=0622 0 0
```

В каждой строке этого файла указано: - имя устройство; - точка монтирования; - тип файловой системы; - опции монтирования; - специальные флаги для утилиты dump; - порядок проверки целостности файловой системы с помощью утилиты fsck.

df

Для определения объёма свободного пространства на файловой системе можно воспользоваться командой `df`, которая выведет на экран список всех файловых систем в соответствии с именами устройств, с указанием размера и точки монтирования.

Например:

df

```
Filesystem 1024-blocks Used Available Capacity Mounted on
/dev/hda3 297635 169499 112764 60% /
```

С помощью команды `fsck` можно проверить (а в ряде случаев восстановить) целостность файловой системы:

Формат команды: `fsck имя_устройства`

Пример:

```
fsck /dev/sda1
```

Выполнение лабораторной работы

1. Выполните все примеры, приведённые в первой части описания лабораторной работы.

Пример №1

1. Копирование файла в текущем каталоге. Скопировать файл `~/abc1` в файл `april` и в файл `may`:

```
[epmitichkina@fedora ~]$ touch abc1
[epmitichkina@fedora ~]$ cp abc1 april
[epmitichkina@fedora ~]$ cp abc1 may
[epmitichkina@fedora ~]$ ls
01  abc1  may  work  Документы  Изображения  Общедоступные  Шаблоны
5   april  newdir  Видео  Загрузки  Музыка  'Рабочий стол'
```

2. Копирование нескольких файлов в каталог. Скопировать файлы `april` и `may` в каталог `monthly`:

```
[epmitichkina@fedora ~]$ mkdir montly
[epmitichkina@fedora ~]$ cp april may montly
[epmitichkina@fedora ~]$ ls
01  abc1  may  newdir  Видео  Загрузки  Музыка  'Рабочий стол'
5   april  montly  work  Документы  Изображения  Общедоступные  Шаблоны
[epmitichkina@fedora ~]$ ls montly/
april may
[epmitichkina@fedora ~]$
```

3. Копирование файлов в произвольном каталоге. Скопировать файл `monthly/may` в файл с именем `june`:

```
[epmitichkina@fedora ~]$ cp montly/may montly/june
[epmitichkina@fedora ~]$ ls montly/
april  june  may
[epmitichkina@fedora ~]$ █
```

Пример №2

1. Копирование каталогов в текущем каталоге. Скопировать каталог montly в каталог montly.00:

```
[epmitichkina@fedora ~]$ mkdir montly.00
[epmitichkina@fedora ~]$ cp -r montly montly.00
[epmitichkina@fedora ~]$ ls
01  abcl  may  montly.00  work  Документы  Изображения  Общедоступные  Шаблоны
5   april  montly  newdir  Видео  Загрузки  Музыка  'Рабочий стол'
```

2. Копирование каталогов в произвольном каталоге. Скопировать каталог montly.00 в каталог /tmp

```
[epmitichkina@fedora ~]$ cp -r montly.00 /tmp
[epmitichkina@fedora ~]$ tree
.
├── 01
│   ├── hugo_extended_0.98.0_Linux-64bit
│   │   ├── LICENSE
│   │   └── README.md
│   └── hugo_extended_0.98.0_Linux-64bit.tar.gz
├── 5
├── abcl
├── april
├── may
├── montly
│   ├── april
│   ├── june
│   └── may
├── montly.00
│   └── montly
│       ├── april
│       ├── june
│       └── may
├── newdir
└── work
```

288 directories, 370 files

```
[epmitichkina@fedora ~]$ cd /tmp
[epmitichkina@fedora tmp]$ ls
montly.00
```

Пример №3

1. Переименование файлов в текущем каталоге. Изменить название файла april на july в домашнем каталоге:

```
[epmitichkina@fedora tmp]$ cd
[epmitichkina@fedora ~]$ mv april july
[epmitichkina@fedora ~]$ ls
01  abcl  may  montly.00  work  Документы  Изображения  Общедоступные  Шаблоны
5   july  montly  newdir  Видео  Загрузки  Музыка  'Рабочий стол'
```

2. Перемещение файлов в другой каталог. Переместить файл july в каталог montly.00:

```

[epmitichkina@fedora ~]$ mv july montly.00
[epmitichkina@fedora ~]$ ls montly.00
july  montly
[epmitichkina@fedora ~]$ █

```

3. Переименование каталогов в текущем каталоге. Переименовать каталог monthly.00 в monthly.01

```

[epmitichkina@fedora ~]$ mv montly.00 montly.01
[epmitichkina@fedora ~]$ ls
01  abc1  montly  newdir  Видео  Загрузки  Музыка  'Рабочий стол'
5   may  montly.01  work  Документы  Изображения  Общедоступные  Шаблоны
[epmitichkina@fedora ~]$ █

```

4. Перемещение каталога в другой каталог. Переместить каталог monthly.01 в каталог reports:

```

[epmitichkina@fedora ~]$ mkdir reports
[epmitichkina@fedora ~]$ mv montly.01 reports
[epmitichkina@fedora ~]$ ls
01  abc1  montly  reports  Видео  Загрузки  Музыка  'Рабочий стол'
5   may  newdir  work  Документы  Изображения  Общедоступные  Шаблоны
[epmitichkina@fedora ~]$ ls reports
montly.01
[epmitichkina@fedora ~]$ █

```

5. Переименование каталога, не являющегося текущим. Переименовать каталог reports/monthly.01 в reports/monthly:

```

montly.01
[epmitichkina@fedora ~]$ mv reports/montly.01 reports/monthly
[epmitichkina@fedora ~]$ ls reports/
montly
[epmitichkina@fedora ~]$ █

```

Пример №4

1. Требуется создать файл ~/may с правом выполнения для владельца:

```

[epmitichkina@fedora ~]$ cd
[epmitichkina@fedora ~]$ touch may
[epmitichkina@fedora ~]$ ls -l may
-rw-rw-r--. 1 epmitichkina epmitichkina 0 мая  4 14:06 may
[epmitichkina@fedora ~]$ chmod u+x may
[epmitichkina@fedora ~]$ ls -l may
-rwxrw-r--. 1 epmitichkina epmitichkina 0 мая  4 14:06 may
[epmitichkina@fedora ~]$ █

```

2. Требуется лишить владельца файла ~/may права на выполнение:

```

[epmitichkina@fedora ~]$ chmod u-x may
[epmitichkina@fedora ~]$ ls -l may
-rw-rw-r--. 1 epmitichkina epmitichkina 0 мая  4 14:06 may
[epmitichkina@fedora ~]$ █

```

3. Требуется создать каталог monthly с запретом на чтение для членов группы и всех остальных пользователей:

```

[epmitichkina@fedora ~]$ mkdir monthly
[epmitichkina@fedora ~]$ chmod g-r, o-r monthly
chmod: неверный режим: «g-r,»
По команде «chmod --help» можно получить дополнительную информацию.
[epmitichkina@fedora ~]$ cd

```

4. Требуется создать файл ~/abc1 с правом записи для членов группы:

по команде «стпюа --пепр» можно получить дополнительную информацию.

```
[epmitichkina@fedora ~]$ cd
[epmitichkina@fedora ~]$ touch abc1
[epmitichkina@fedora ~]$ chmod g+w abc1
[epmitichkina@fedora ~]$ ls -l may
-rw-rw-r--. 1 epmitichkina epmitichkina 0 мая  4 14:06 may
[epmitichkina@fedora ~]$
```

Пример №5

1. Для просмотра используемых в операционной системе файловых систем можно воспользоваться командой `mount` без параметров. В результате её применения можно получить примерно следующее:

```
[epmitichkina@fedora ~]$ mount
proc on /proc type proc (rw,nosuid,nodev,noexec,relatime)
sysfs on /sys type sysfs (rw,nosuid,nodev,noexec,relatime,seclabel)
devtmpfs on /dev type devtmpfs (rw,nosuid,seclabel,size=986252k,nr_inodes=246563,mode=755,inode64)
securityfs on /sys/kernel/security type securityfs (rw,nosuid,nodev,noexec,relatime)
tmpfs on /dev/shm type tmpfs (rw,nosuid,nodev,seclabel,inode64)
devpts on /dev/pts type devpts (rw,nosuid,noexec,relatime,seclabel,gid=5,mode=620,ptmxmode=000)
tmpfs on /run type tmpfs (rw,nosuid,nodev,seclabel,size=402660k,nr_inodes=819200,mode=755,inode64)
cgroup2 on /sys/fs/cgroup type cgroup2 (rw,nosuid,nodev,noexec,relatime,seclabel,nsdelegate,memory_recursivepr
pstore on /sys/fs/pstore type pstore (rw,nosuid,nodev,noexec,relatime,seclabel)
none on /sys/fs/bpf type bpf (rw,nosuid,nodev,noexec,relatime,mode=700)
/dev/sda2 on / type btrfs (rw,relatime,seclabel,compress=zstd:1,space_cache,subvol=258,subvol=/root)
selinuxfs on /sys/fs/selinux type selinuxfs (rw,nosuid,noexec,relatime)
systemd-1 on /proc/sys/fs/binfmt_misc type autofs (rw,relatime,fd=31,pgrp=1,timeout=0,minproto=5,maxproto=5,dia
,pipe_ino=15332)
mqueue on /dev/mqueue type mqueue (rw,nosuid,nodev,noexec,relatime,seclabel)
hugetlbfs on /dev/hugepages type hugetlbfs (rw,relatime,seclabel,pagesize=2M)
debugfs on /sys/kernel/debug type debugfs (rw,nosuid,nodev,noexec,relatime,seclabel)
tracefs on /sys/kernel/tracing type tracefs (rw,nosuid,nodev,noexec,relatime,seclabel)
configfs on /sys/kernel/config type configfs (rw,nosuid,nodev,noexec,relatime)
fusectl on /sys/fs/fuse/connections type fusectl (rw,nosuid,nodev,noexec,relatime)
tmpfs on /tmp type tmpfs (rw,nosuid,nodev,seclabel,nr_inodes=409600,inode64)
/dev/sda2 on /home type btrfs (rw,relatime,seclabel,compress=zstd:1,space_cache,subvol=256,subvol=/home)
/dev/sda1 on /boot type ext4 (rw,relatime,seclabel)
sunrpc on /var/lib/nfs/rpc_pipefs type rpc_pipefs (rw,relatime)
tmpfs on /run/user/1000 type tmpfs (rw,nosuid,nodev,relatime,seclabel,size=201328k,nr_inodes=50332,mode=700,uid
00,gid=1000,inode64)
gvfsd-fuse on /run/user/1000/gvfs type fuse.gvfsd-fuse (rw,nosuid,nodev,relatime,user_id=1000,group_id=1000)
[epmitichkina@fedora ~]$
```

2. Другой способ определения смонтированных в операционной системе файловых систем — просмотр файла `/etc/fstab`. Сделать это можно например с помощью команды `cat`:

```
[epmitichkina@fedora ~]$ cat /etc/fstab
#
# /etc/fstab
# Created by anaconda on Tue Apr 19 08:28:21 2022
#
# Accessible filesystems, by reference, are maintained under '/dev/disk/'.
# See man pages fstab(5), findfs(8), mount(8) and/or blkid(8) for more info.
#
# After editing this file, run 'systemctl daemon-reload' to update systemd
# units generated from this file.
#
UUID=f7c88261-8ed7-4a23-b3fb-a6c4f7855fa1 / btrfs subvol=root,compress=zstd:1 0 0
UUID=1c0004a5-f17b-4a32-85ba-57510f5aafe8 /boot ext4 defaults 1 2
UUID=f7c88261-8ed7-4a23-b3fb-a6c4f7855fa1 /home btrfs subvol=home,compress=zstd:1 0 0
[epmitichkina@fedora ~]$
```

3. Для определения объёма свободного пространства на файловой системе можно воспользоваться командой `df`, которая выведет на экран список всех файловых систем в соответствии с именами устройств, с указанием размера и точки монтирования.


```
[epmitichkina@fedora ~]$ df
Файловая система 1К-блоков  Использовано  Доступно  Использовано%  Смонтировано в
devtmpfs          986252      0    986252          0% /dev
tmpfs             1006644    14812   991832          2% /dev/shm
tmpfs             402660     1376   401284          1% /run
/dev/sda2         82836480   5608112 75506784         7% /
tmpfs            1006644      68   1006576          1% /tmp
/dev/sda2         82836480   5608112 75506784         7% /home
/dev/sda1         996780    175308   752660         19% /boot
tmpfs            201328     132    201196          1% /run/user/1000
[epmitichkina@fedora ~]$
```

4. С помощью команды fsck можно проверить (а в ряде случаев восстановить) целостность файловой системы:

```
tmpfs            201328     132    201196          1% /run/user/1000
[epmitichkina@fedora ~]$ fsck /dev/sda2
fsck из util-linux 2.37.2
If you wish to check the consistency of a BTRFS filesystem or
repair a damaged filesystem, see btrfs(8) subcommand 'check'.
[epmitichkina@fedora ~]$ fsck /dev/sda1
```

2. Выполните следующие действия, зафиксировав в отчёте по лабораторной работе используемые при этом команды и результаты их выполнения:

1. Скопируйте файл /usr/include/sys/io.h в домашний каталог и назовите его equipment. Если файла io.h нет, то используйте любой другой файл в каталоге /usr/include/sys/ вместо него.

```
[epmitichkina@fedora ~]$ cp /usr/include/sys/io.h ~/equipment
[epmitichkina@fedora ~]$ ls
01  abc1    may      montly  reports  Видео    Загрузки  Музыка    'Рабочий стол'
5   equipment  monthly  newdir  work     Документы  Изображения  Общедоступные  Шаблоны
[epmitichkina@fedora ~]$ cat equipment
/* Copyright (C) 1996-2021 Free Software Foundation, Inc.
   This file is part of the GNU C Library.

   The GNU C Library is free software; you can redistribute it and/or
   modify it under the terms of the GNU Lesser General Public
   License as published by the Free Software Foundation; either
   version 2.1 of the License, or (at your option) any later version.

   The GNU C Library is distributed in the hope that it will be useful,
   but WITHOUT ANY WARRANTY; without even the implied warranty of
   MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
   Lesser General Public License for more details.

   You should have received a copy of the GNU Lesser General Public
   License along with this library; if not, see <http://www.gnu.org/licenses/old-licenses/lgpl-2.1.html>.
```

2. В домашнем каталоге создайте директорию ~/ski.places.

```
[epmitichkina@fedora ~]$ mkdir ski.places
[epmitichkina@fedora ~]$ ls
01  abc1    may      montly  reports  work     Документы  Изображения  Общедоступные  Шаблоны
5   equipment  monthly  newdir  ski.places  Видео    Загрузки  Музыка    'Рабочий стол'
```

3. Переместите файл equipment в каталог ~/ski.places.

```
[epmitichkina@fedora ~]$ mv equipment ski.places
[epmitichkina@fedora ~]$ ls
01  abc1    monthly  newdir  ski.places  Видео    Загрузки  Музыка    'Рабочий стол'
5   may     montly   reports  work     Документы  Изображения  Общедоступные  Шаблоны
[epmitichkina@fedora ~]$ ls ski.places/
equipment
```

4. Переименуйте файл ~/ski.places/equipment в ~/ski.places/equiplist.

```
[epmitichkina@fedora ~]$ mv ski.places/equipment ski.places/equiplist
[epmitichkina@fedora ~]$ ls ski.places/
equiplist
```

5. Создайте в домашнем каталоге файл `abc1` и скопируйте его в каталог `~/ski.places`, назовите его `equiplist2`.

```
[epmitichkina@fedora ~]$ touch abc1
[epmitichkina@fedora ~]$ cp abc1 ski.places/equiplist2
[epmitichkina@fedora ~]$ ls ski.places/
equiplist equiplist2
```

6. Создайте каталог с именем `equipment` в каталоге `~/ski.places`.

```
[epmitichkina@fedora ~]$ cd ski.places/
[epmitichkina@fedora ski.places]$ mkdir equipment
```

7. Переместите файлы `~/ski.places/equiplist` и `equiplist2` в каталог `~/ski.places/equipment`.

```
[epmitichkina@fedora ski.places]$ mv equiplist equipment
[epmitichkina@fedora ski.places]$ mv equiplist2 equipment
[epmitichkina@fedora ski.places]$ ls equipment/
equiplist equiplist2
[epmitichkina@fedora ski.places]$ ls
equipment
```

8. Создайте и переместите каталог `~/newdir` в каталог `~/ski.places` и назовите его `plans`.

```
[epmitichkina@fedora ~]$ rmdir newdir
[epmitichkina@fedora ~]$ mkdir newdir
[epmitichkina@fedora ~]$ mv newdir ski.places/plans
[epmitichkina@fedora ~]$ ls ski.places/
equipment plans
```

3. Определите опции команды `chmod`, необходимые для того, чтобы присвоить перечисленным ниже файлам выделенные права доступа, считая, что в начале таких прав нет:

1. `drwxr--r-- ... australia`

```
[epmitichkina@fedora ~]$ touch australia
[epmitichkina@fedora ~]$ chmod 744 australia
[epmitichkina@fedora ~]$ ls -l australia
-rwxr--r--. 1 epmitichkina epmitichkina 0 мая  4 14:35 australia
[epmitichkina@fedora ~]$
```

2. `drwx--x--x ... play`

```
[epmitichkina@fedora ~]$ touch play
[epmitichkina@fedora ~]$ chmod 711 play
[epmitichkina@fedora ~]$ ls -l play
-rwx--x--x. 1 epmitichkina epmitichkina 0 мая  4 14:37 play
[epmitichkina@fedora ~]$
```

3. `-r-xr--r-- ... my_os`

```
[epmitichkina@fedora ~]$ touch my_os
[epmitichkina@fedora ~]$ chmod 544 my_os
[epmitichkina@fedora ~]$ ls -l my_os
-r-xr--r--. 1 epmitichkina epmitichkina 0 мая  4 14:42 my_os
[epmitichkina@fedora ~]$
```

4. `-rw-rw-r-- ... feathers`

```
[epmitichkina@fedora ~]$ touch feathers
[epmitichkina@fedora ~]$ chmod 664 feathers
[epmitichkina@fedora ~]$ ls -l feathers
-rw-rw-r--. 1 epmitichkina epmitichkina 0 мая  4 14:43 feathers
[epmitichkina@fedora ~]$
```

4. Прodelайте приведённые ниже упражнения, записывая в отчёт по лабораторной работе используемые при этом команды:

4.1. Просмотрите содержимое файла /etc/password.

```
[epmitichkina@fedora ~]$ cat /etc/passwd
cat: /etc/passwd: Нет такого файла или каталога
[epmitichkina@fedora ~]$
```

4.2. Скопируйте файл ~/feathers в файл ~/file.old.

```
cat: /etc/passwd: Нет такого файла или каталога
[epmitichkina@fedora ~]$ cp feathers file.old
[epmitichkina@fedora ~]$ ls
01  australia  may      my_os  ski.places  Документы  Музыка  Шаблоны
5   feathers    monthly  play   work        Загрузки  Общедоступные
abc1 file.old    montly   reports Видео       Изображения  'Рабочий стол'
```

4.3. Переместите файл ~/file.old в каталог ~/play.

```
[epmitichkina@fedora ~]$ mkdir play
[epmitichkina@fedora ~]$ cp file.old play
[epmitichkina@fedora ~]$ ls play/
file.old
[epmitichkina@fedora ~]$
```

4.4. Скопируйте каталог ~/play в каталог ~/fun.

```
[epmitichkina@fedora ~]$ cp -r ~/play ~/fun
[epmitichkina@fedora ~]$ ls
01  australia  fun      montly  reports  Видео  Изображения  'Рабочий стол'
5   feathers    may      my_os   ski.places  Документы  Музыка  Шаблоны
abc1 file.old    monthly  play    work      Загрузки  Общедоступные
```

4.5. Переместите каталог ~/fun в каталог ~/play и назовите его games.

```
abc1 file.old  montly  play  work  Загрузки  Общедоступные
[epmitichkina@fedora ~]$ mv ~/fun ~/play/games
[epmitichkina@fedora ~]$ cd play/
[epmitichkina@fedora play]$ ls
file.old  games
[epmitichkina@fedora play]$ ls games/
file.old
[epmitichkina@fedora play]$
```

4.6. Лишите владельца файла ~/feathers права на чтение.

```
[epmitichkina@fedora ~]$ chmod u-r feathers
[epmitichkina@fedora ~]$ ls -l feathers
--w-rw-r--. 1 epmitichkina epmitichkina 0 мая  4 14:43 feathers
[epmitichkina@fedora ~]$
```

4.7. Что произойдёт, если вы попытаетесь просмотреть файл ~/feathers командой cat?

```
--w-rw-r--. 1 epmitichkina epmitichkina 0 мая  4 14:43 feathers
[epmitichkina@fedora ~]$ cat feathers
cat: feathers: Отказано в доступе
[epmitichkina@fedora ~]$
```

4.8. Что произойдёт, если вы попытаетесь скопировать файл ~/feathers?

```
cat: feathers: Отказано в доступе
[epmitichkina@fedora ~]$ cp feathers feathers2
cp: невозможно открыть 'feathers' для чтения: Отказано в доступе
[epmitichkina@fedora ~]$
```

4.9. Дайте владельцу файла ~/feathers право на чтение.

```
[epmitichkina@fedora ~]$ chmod u+r feathers
[epmitichkina@fedora ~]$ ls -l feathers
-rw-rw-r--. 1 epmitichkina epmitichkina 0 мая  4 14:43 feathers
[epmitichkina@fedora ~]$
```

4.10. Лишите владельца каталога ~/play права на выполнение.

```
[epmitichkina@fedora ~]$ chmod u-x play
[epmitichkina@fedora ~]$ ls -l play/
ls: невозможно получить доступ к 'play/file.old': Отказано в доступе
ls: невозможно получить доступ к 'play/games': Отказано в доступе
итого 0
-???????? ? ? ? ? ? ? file.old
d???????? ? ? ? ? ? ? games
[epmitichkina@fedora ~]$
```

4.11. Перейдите в каталог ~/play. Что произошло?

```
[epmitichkina@fedora ~]$ cd play/
bash: cd: play/: Отказано в доступе
[epmitichkina@fedora ~]$
```

4.12. Дайте владельцу каталога ~/play право на выполнение.

```
[epmitichkina@fedora ~]$ chmod u+x play
[epmitichkina@fedora ~]$ ls -l play/
итого 0
-rw-rw-r--. 1 epmitichkina epmitichkina 0 мая 4 14:51 file.old
drwxrwxr-x. 1 epmitichkina epmitichkina 16 мая 4 14:52 games
[epmitichkina@fedora ~]$
```

5. Прочитайте man по командам mount, fsck, mkfs, kill и кратко их охарактеризуйте, приведя примеры.

1. **mount** монтирует запоминающее устройство или файловую систему, делая их доступными и присоединяя к существующей структуре каталогов

```
NAME
    mount - mount a filesystem

SYNOPSIS
    mount [-h|-V]

    mount [-l] [-t fstype]

    mount -a [-fFnrsvw] [-t fstype] [-O optlist]

    mount [-fnrsvw] [-o options] device|mountpoint

    mount [-fnrsvw] [-t fstype] [-o options] device mountpoint

    mount --bind|--rbind|--move olddir newdir

    mount --make-[shared|slave|private|unbindable|rshared|rslave|rprivate|runbindable] mountpoint

DESCRIPTION
    All files accessible in a Unix system are arranged in one big tree, the file hierarchy, rooted at /.
    These files can be spread out over several devices. The mount command serves to attach the filesystem
    found on some device to the big file tree. Conversely, the umount(8) command will detach it again. The
    filesystem is used to control how data is stored on the device or provided in a virtual way by network
    or other services.

    The standard form of the mount command is:

        mount -t type device dir

    This tells the kernel to attach the filesystem found on device (which is of type type) at the directory
    dir. The option -t type is optional. The mount command is usually able to detect a filesystem. The root
    permissions are necessary to mount a filesystem by default. See section "Non-superuser mounts" below for
    more details. The previous contents (if any) and owner and mode of dir become invisible, and as long as
    this filesystem remains mounted, the pathname dir refers to the root of the filesystem on device.

    If only the directory or the device is given, for example:

        mount /dir

    then mount looks for a mountpoint (and if not found then for a device) in the /etc/fstab file. It's
    possible to use the --target or --source options to avoid ambiguous interpretation of the given
    argument. For example:

        mount --target /mountpoint

Manual page mount(8) line 1 (press h for help or q to quit)
```

2. **fsck** утилита командной строки, которая позволяет выполнять проверки согласованности и интерактивное исправление в одной или нескольких файловых

системах Linux.

NAME

fsck - check and repair a Linux filesystem

SYNOPSIS

fsck [-lsAVRTMNP] [-r [fd]] [-C [fd]] [-t fstype] [filesystem...] [--] [fs-specific-options]

DESCRIPTION

fsck is used to check and optionally repair one or more Linux filesystems. filesystem can be a device name (e.g., /dev/hdc1, /dev/sdb2), a mount point (e.g., /, /usr, /home), or an filesystem label or UUID specifier (e.g., UUID=8868abf6-88c5-4a83-98b8-bfc24057f7bd or LABEL=root). Normally, the **fsck** program will try to handle filesystems on different physical disk drives in parallel to reduce the total amount of time needed to check all of them.

If no filesystems are specified on the command line, and the **-A** option is not specified, **fsck** will default to checking filesystems in /etc/fstab serially. This is equivalent to the **-As** options.

The exit status returned by **fsck** is the sum of the following conditions:

- 0**
No errors
- 1**
Filesystem errors corrected
- 2**
System should be rebooted
- 4**
Filesystem errors left uncorrected
- 8**
Operational error
- 16**
Usage or syntax error
- 32**
Checking canceled by user request
- 128**
Shared-library error

The exit status returned when multiple filesystems are checked is the bit-wise OR of the exit statuses for each filesystem that is checked.

3. **mkfs** создаёт новую файловую систему Linux

```
NAME
mkfs - build a Linux filesystem

SYNOPSIS
mkfs [options] [-t type] [fs-options] device [size]

DESCRIPTION
This mkfs frontend is deprecated in favour of filesystem specific mkfs.<type> utils.

mkfs is used to build a Linux filesystem on a device, usually a hard disk partition. The device argument is either the device name (e.g., /dev/hda1, /dev/sdb2), or a regular file that shall contain the filesystem. The size argument is the number of blocks to be used for the filesystem.

The exit status returned by mkfs is 0 on success and 1 on failure.

In actuality, mkfs is simply a front-end for the various filesystem builders (mkfs.fstype) available under Linux. The filesystem-specific builder is searched for via your PATH environment setting only. Please see the filesystem-specific builder manual pages for further details.

OPTIONS
-t, --type type
    Specify the type of filesystem to be built. If not specified, the default filesystem type (current ext2) is used.

fs-options
    Filesystem-specific options to be passed to the real filesystem builder.

-V, --verbose
    Produce verbose output, including all filesystem-specific commands that are executed. Specifying this option more than once inhibits execution of any filesystem-specific commands. This is really only useful for testing.

-V, --version
    Display version information and exit. (Option -V will display version information only when it is the only parameter, otherwise it will work as --verbose.)

-h, --help
    Display help text and exit.

BUGS
All generic options must precede and not be combined with filesystem-specific options. Some filesystem-specific programs do not automatically detect the device size and require the size parameter to be specified.

AUTHORS
Manual page mkfs(8) line 1 (press h for help or q to quit)
```

4. **kill** посылает сигнал процессу или выводит список допустимых сигналов

```
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 not give the target process the opportunity to perform any clean-up before terminating.

    Most modern shells have a builtin kill command, with a usage rather similar to that of the command described here. The --all, --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 of the form '-n' is given, and it is meant to denote a process group, either a signal must be specified first, or the argument must be preceded by a '--' option, otherwise it will be taken as the signal to send.

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

Выводы

В результате работы ознакомилась с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобретил практических навыков по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

Ответы на контрольные вопросы

1. Дайте характеристику каждой файловой системе, существующей на жёстком диске компьютера, на котором вы выполняли лабораторную работу.
Tmpfs - временное файловое хранилище во многих Unix-подобных ОС и в частности Linux. Предназначена для монтирования файловой системы, но размещается в ОЗУ вместо физического диска.

2. Приведите общую структуру файловой системы и дайте характеристику каждой директории первого уровня этой структуры.
- `/bin` - каталог, содержащий исполняемые файлы. Монтируется на корневую файловую систему, должен быть доступен даже если никакие другие файловые системы не смонтированы.
 - `/dev` - содержит файлы физических устройств, которые могут входить в состав аппаратного обеспечения компьютера.
 - `/home` - каталог, содержащий в себе домашние каталоги пользователей операционной системы, в которых хранятся их данные, настройки, пароли и т.д.
 - `/lib64` - каталог, присутствующий на 64-битных системах, содержащий набор библиотек и компонентов компилятора языка C для 64-битных систем.
 - `/media` - точка для автоматического монтирования различных устройств: USB-накопители, CD-ROM и т.д.
 - `/opt` - содержит подкаталоги для дополнительных пакетов программного обеспечения.
 - `/root` - Домашний каталог пользователя root. Он мог бы лежать в папке `/home`, но находится на первом уровне для большей надежности системы.
 - `/sbin` - содержит исполняемые файлы, предназначенные для запуска пользователем при администрировании системы.
 - `/sys` - точка монтирования виртуальной файловой системы `sysfs` с информацией об устройствах, драйверах, ядре ОС, гипервизоре и т.д.
 - `/usr` - бинарные файлы, используемые только пользователями, например игры.
 - `/boot` - файлы, нужные для запуска ОС (образы ядер Linux и файлы менеджеров загрузки).
 - `/etc` - содержит основные конфигурационные файлы операционной системы и различных программ.
 - `/lib` - директория, предназначенная для хранения системных библиотек и компонентов компилятора языка C, необходимых для работы программ из каталогов `/bin` и `/sbin`.
 - `/lost+found` - При сбое в работе файловой системы и дальнейшей проверки файлов все найденные поврежденные файлы помещаются в каталог `/lost+found`, чтобы их можно было попытаться восстановить.
 - `/mnt` - Точка ручного монтирования (используется для временного монтирования с применением команды `mount`).

- /proc - содержит файлы ядра и процессора. В эту директорию примонтирована виртуальная файловая система procfs, в которой содержатся специальные файлы, в которых находится информация о системе и выполняющихся процессах.
 - /run - каталог для хранения вспомогательных временных файлов приложений.
 - /srv - содержит данные сервисных служб, предоставляемых системой.
 - /tmp - содержит временные файлы, которые удаляются при выключении или перезагрузке системы.
 - /var - содержит журналы ОС, системные логи и cache-файлы.
3. Какая операция должна быть выполнена, чтобы содержимое некоторой файловой системы было доступно операционной системе?
mkfs
 4. Назовите основные причины нарушения целостности файловой системы. Как устранить повреждения файловой системы?

Основной причиной нарушения целостности файловой системы является остановка ОС в момент обновления метаданных файлов. Это может привести к дублированию или наоборот потере файлов из-за нарушения инварианта счетчика ссылок или другой важной информации. Для диагностики и исправления ошибок файловой системы используется команда fsck.

5. Как создаётся файловая система?
Создать новую файловую систему можно с помощью команды mkfs.
6. Дайте характеристику командам для просмотра текстовых файлов.
 - cat - подходит для просмотра файлов небольшого размера, выводит всё их содержимое в консоль.
 - less - подходит для постраничного просмотра файла, можно переключаться между страницами.
 - head - выводит несколько первых строк файла, по умолчанию - 10, но можно указать и другое число.
 - tail - выводит несколько последних строк файла, по умолчанию - 10, но можно указать и другое число.
7. Приведите основные возможности команды cp в Linux.
Команда cp применяется для копирования файлов и каталогов, имеется возможность копирования несколько файлов в один каталог одной командой. Можно копировать файлы в произвольном каталоге, указывая полный или при возможности относительный путь. Для копирования каталогов вместе с их содержимым указывается опция -r. Подробнее об этой команде написано в теоретическом введении.
8. Приведите основные возможности команды mv в Linux.
Команда mv применится для перемещения и смены имени файлов и каталогов. Подробнее об этой команде написано в теоретическом введении.

9. Что такое права доступа? Как они могут быть изменены?

Права доступа - набор значений, устанавливающий возможности тех или иных пользователей читать, изменять или исполнять конкретный файл или директорию. Права доступа к файлу можно изменить с помощью команды `chmod`. Подробнее об этой команде написано в теоретическом введении.