# РОCСИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

## Факультет физико-математических и естественных наук

## Кафедра прикладной информатики и теории вероятностей

## ОТЧЕТ

## ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5

### *дисциплина: Операционные системы*

Студент: Губина Ольга Вячеславовна Группа: НПИбд-01-20

Преподаватель: Велиева Татьяна Рефатовна

МОСКВА

2021 г.

### Цель работы:

Приобретение практических навыков взаимодействия пользователя с системой посредством командной строки.

### Теоретические сведения:

В данной лаботаторной работе нам придется в заимодействовать непосредственно с командой строкой, поэтому предварительно нам необходимо ознакомиться с *основными командами*[[1]](https://www.hostinger.ru/rukovodstva/osnovnyje-komandy-linux) консоли и их синтаксисом. Далее будет приведена подробная информация об использовании командной строки.

**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

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

Формат команды: cd [путь\_к\_каталогу]

Для перехода в домашний каталог пользователя следует использовать команду cd без параметров или 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:

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

Данные команды были приведены в качестве примера для ознакомления с синтаксисом. Подробнее ознакомиться со всеми командами системы можно по ссылке указанной выше ([[1]](https://www.hostinger.ru/rukovodstva/osnovnyje-komandy-linux)) или в библиографии.

### Выполнение работы:

1. Нам нужно определить полное имя домашнего каталога, далее относительно которого будут выполняться последующие действия.

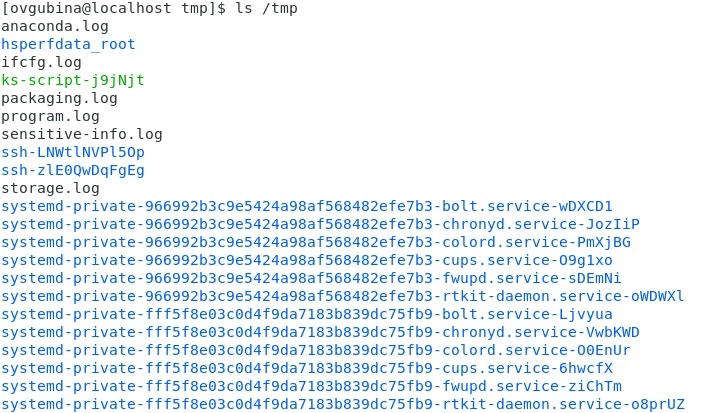
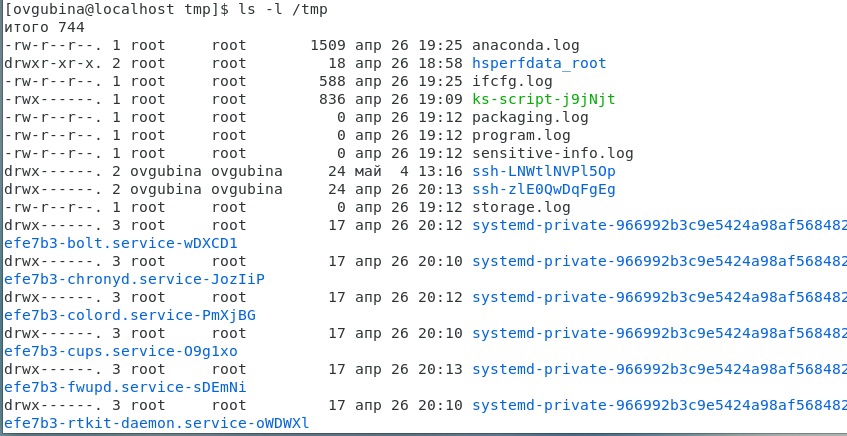
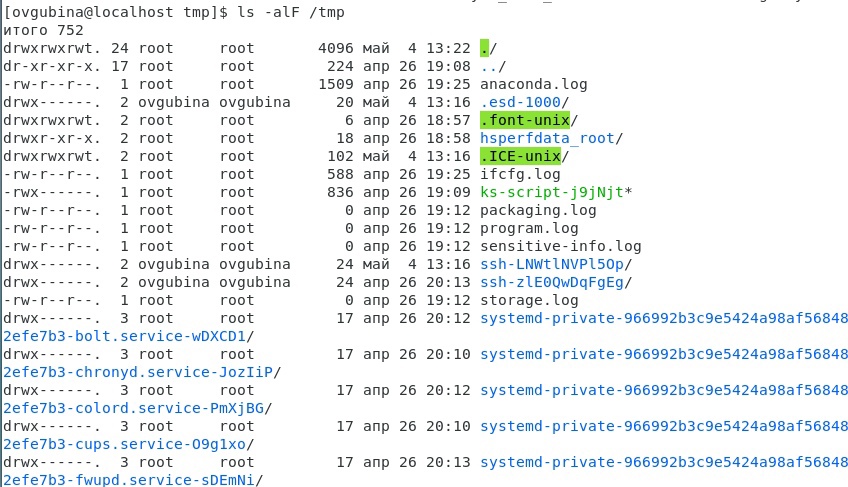
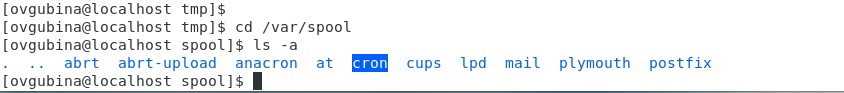
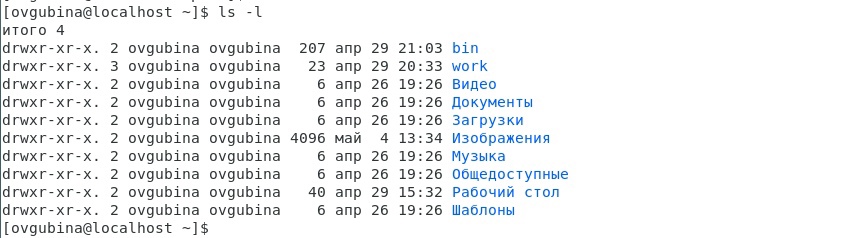
Для этого сначала перейдем в домашний каталог командой cd, а после воспользуемся командой pwd, позволяющей узнать полное имя текущего каталога (*рисунок 1*).

*Рисунок 1: полное имя домашнего каталога:*

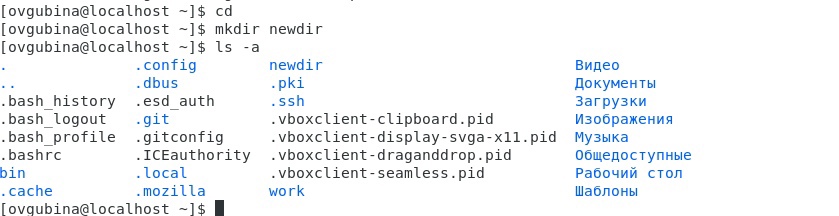
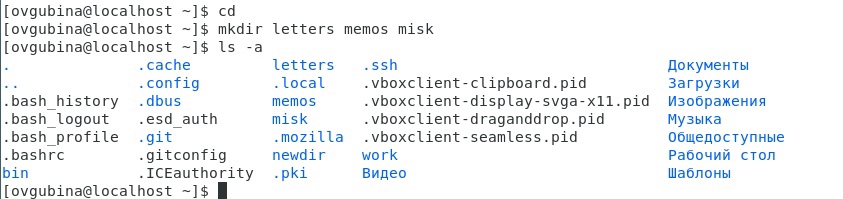
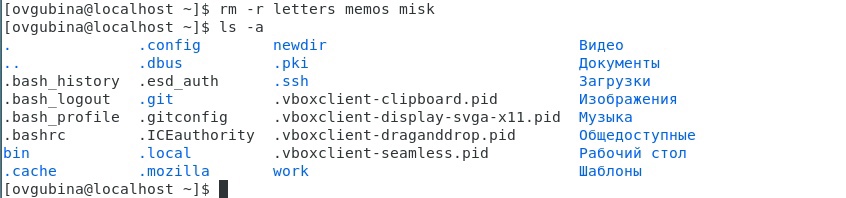
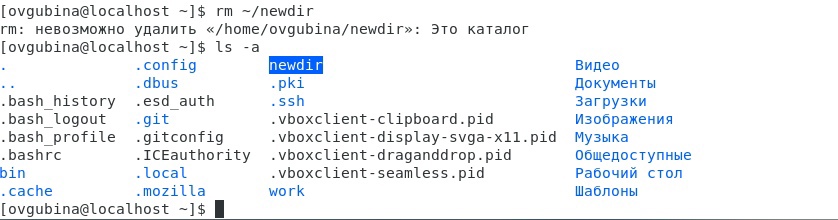
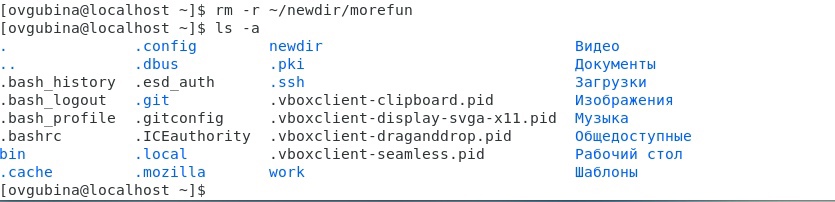


Видим, что домашний каталог имеет полное имя /home/ovgubina, где ovgubina - имя пользователя.

1. Выполняем следующие действия:

* 2.1. Переходим в каталог /tmp, воспользовавшись командой cd /tmp (*рисунок 2*).
* *Рисунок 2: переход в каталог /tmp*
* 
* 2.2. Выводим на экран содержимое каталога /tmp командой ls с различными опциями.
* Сперва вводим ls без опций (*рисунок 3*), это позволит нам увидеть содержимое каталога без скрытых файлов, их типов и т.д., то есть, мы увидим лишь названия файлов и каталогов, хранящихся в каталоге /tmp.
* *Рисунок 3: вывод команды ls*
* 
* Теперь просмотрим содержимое каталога вызовом команды ls -a (*риуснок 4*). Таким образом мы увидим еще и скрытые файлы (они подсвечены зеленым), которые используются для настройки среды.
* *Рисунок 4: вызов команды ls -a*
* 
* Просмотрим каталог, воспользовавшись командой ls -F (*рисунок 5*), которая выведет на экран содержащиеся в каталоге файлы с обозначением их типов:
  + / - каталог;
  + \* - исполняемый файл;
  + @ - ссылка.
* *Рисунок 5: вывод команды ls -F*
* 
* Далее введем команду ls -l (*рисунок 6*), которая позволит нам просмотреть подробную информацию о каждом файле каталога, а именно: его тип, право доступа, число ссылок, владельца, размер, дату последней ревизии, имя файла или каталога.
* *Рисунок 6: вывод команды ls -l*
* 
* Мы также можем воспользоваться командой ls -alF (*рисунок 7*), чтобы объединить все вышеперечисленные команды. Тогда на выходе мы получим список всех файлов, даже скрытых как при -a, с обозначением их типа как в -F и с подробной информацией как при -l.
* *Рисунок 7: вывод ls -alF*
* 
* 2.3. Теперь нам нужно определить, есть ли в каталоге /var/spool подкаталог с именем cron. Для этого командой cd /var/spool перейдем в данный каталог и выведем его содержимое с помощью ls -a (*рисунок 8*).
* *Рисунок 8: поиск подкаталога cron в каталоге /var/spool*
* 
* Видим, что подкаталог cron действительно находится в данном подкаталоге (он подсвечен синим).
* 2.4. Теперь перйдем в наш домашний каталог (cd) и выведим на экран его содержимое (ls -l, чтобы увидеть подробную информацию о файлах, в том числе и об их пользователе) (*рисунок 9*).
* *Рисунок 9: содержимое домашнего каталога*
* 
* Видим, что владельцем файлов и подкаталогов является пользователь ovgubina, то есть я.

1. Выполняем следующие действия:

* 3.1. В домашнем каталоге создаем новый каталог с именем newdir (*рисунок 10*). Для этого воспользуемся командой mkdir newdir, предварительно перейдя в домашний каталог - cd. После проверяем, создали ли мы его, просматривая содержимое домашнего каталога командой ls -a.
* *Рисунок 10: создание каталога newdir*
* 
* Видим, что каталог был создан успешно.
* 3.2. Перейдем в каталог ~/newdir (cd ~/newdir) и создадим в нем подкаталог morefun (mkdir morefun), после чего проверим верность наших действий (ls -a) (*рисунок 11*).
* *Рисунок 11: создание подкаталога morefun в каталоге ~/newdir*
* 
* Видим, что подкаталог действительно был создан.
* 3.3. Переходим в домашний каталог - cd. В домашнем каталоге создаем одной командой три новых каталога с именами letters, memos, misk - mkdir letters memos misk, и проверяем, просматривая домашний каталог ls -a (*рисунок 12*). Затем удаляем эти каталоги одной командой - rm -r letters memos misk, проверяем - ls -a (*рисунок 13*).
* *Рисунок 12: создание каталогов letters, memos, misk*
* 
* *Рисунок 13: создание каталогов letters, memos, misk*
* 
* Смотрим на результаты команды ls -a и видим, что все выполнено верно.
* 3.4. Пробуем удалить ранее созданный каталог ~/newdir командой rm. Проверяем, был ли каталог удалён - ls -a (*рисунок 14*).
* *Рисунок 14: попытка удалить каталог ~/newdir*
* 
* Видим, что каталог удалить не удалось. Произошло это потому, что мы использовали команду удаления без ключей, в таком случае она является командой удаления файлов, а иы пытались удалить каталог.
* 3.5. Удаляем каталог ~/newdir/morefun из домашнего каталога - rm -r ~/newdir/morefun. Проверяем, был ли каталог удалён, просматриваем домашний каталог - ls -a. (*рисунок 15*)
* *Рисунок 15: удаление каталога ~/newdir/morefun из домашнего каталога*
* 
* Видим что каталог morefun был успешно удален.

1. С помощью команды man ls определяем, какую опцию команды ls нужно использовать для просмотра содержимого не только указанного каталога, но и подкаталогов, входящих в него (*рисунок 16*).

*Рисунок 16: использование man для получения информации о ls*



Простматриваем результат выполнения команды man и находим нужный нам ключ, позволяющий просматривать содержимое каталога и его подкаталогов одновременно. Это рекурсивный просмотр - опция -R (*рисунок 17*).

*Рисунок 17: рекурсивный просмотр; опция -R*



1. Теперь с помощью все той же команды man определяем набор опций команды ls, позволяющий отсортировать по времени последнего изменения выводимый список содержимого каталога с развёрнутым описанием файлов. Просматривем свывод команды man из *пункта 4* и находим искомые опции.

Это опция -l, позволяющая вывести список содержимого каталога с развёрнутым описанием файлов (*рисунок 18*), и опция -t - сортировка по времени последнего изменения выводимого списка содержимого каталога (*рисунок 19*).

*Рисунок 18: опция `-l*



*Рисунок 19: опция -t*

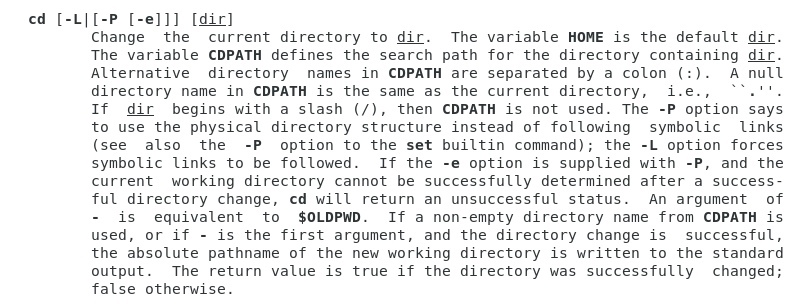


1. Далее используем команду man для просмотра описания следующих команд: cd, pwd, mkdir, rmdir, rm (*рисунок 20*) и просматриваем рузультаты каждой из них (*рисунки 21-26*).

*Рисунок 20: man для просмотра описания cd, pwd, mkdir, rmdir, rm*



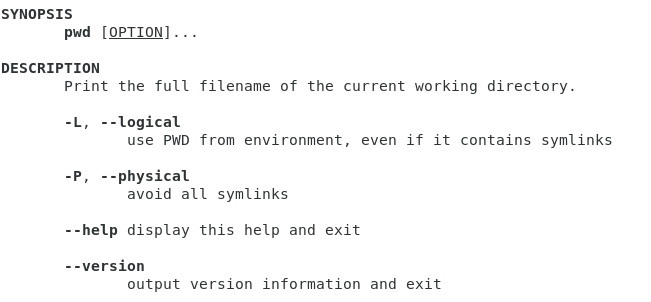
*Рисунок 21: вывод man для cd*



Таким образом для **cd** основные опции это (*рисунок 21*):

* -L - переходит по символическим ссылкам только после того, как были обработаны переходы на одну директорию выше "..";
* -R - позволяет следовать по символическим ссылкам перед тем, как будут обработаны все переходы "..";
* -e - если папку, в которую нужно перейти не удалось найти - выдает ошибку.

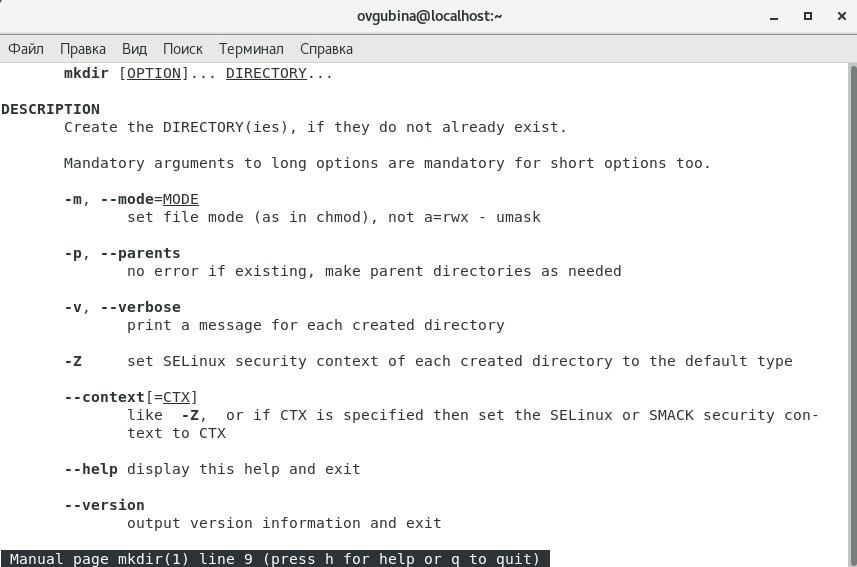
*Рисунок 22: Вывод man для pwd*



Таким образом для **pwd** основные опции это (*рисунок 22*):

* -L, --logical - берет директорию из переменной окружения, даже если она содержит символические ссылки;
* -P, --physycal - отбрасывает все символические ссылки;
* --help - отображает справку по утилите;
* --version - отображает версию утилиты.

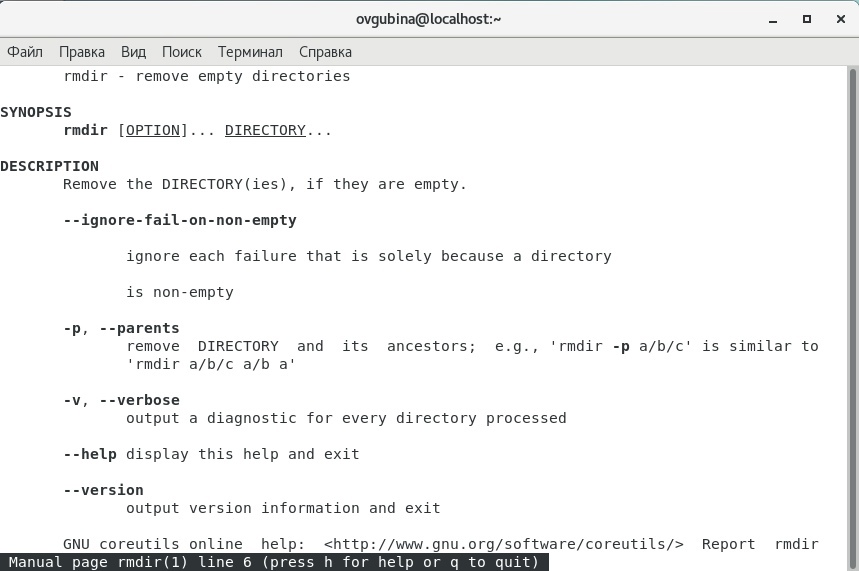
*Рисунок 23: вывод man для mkdir*



Таким образом для **mkdir** основные опции это (*рисунок 23*):

* -m - назначить режим доступа (права). По умолчанию mod принимает значение 0777, что обеспечивает неограниченные права;
* -p - не показывать ошибки, а также их игнорировать;
* -v - выводить сообщение о каждом новым каталоге;
* -z - принимает контекст SELinux для каталога по умолчанию;
* --context - то же, что и -z;
* --help - вывести справочную информацию;
* --version - выводит информацию о текущей версии утилиты.

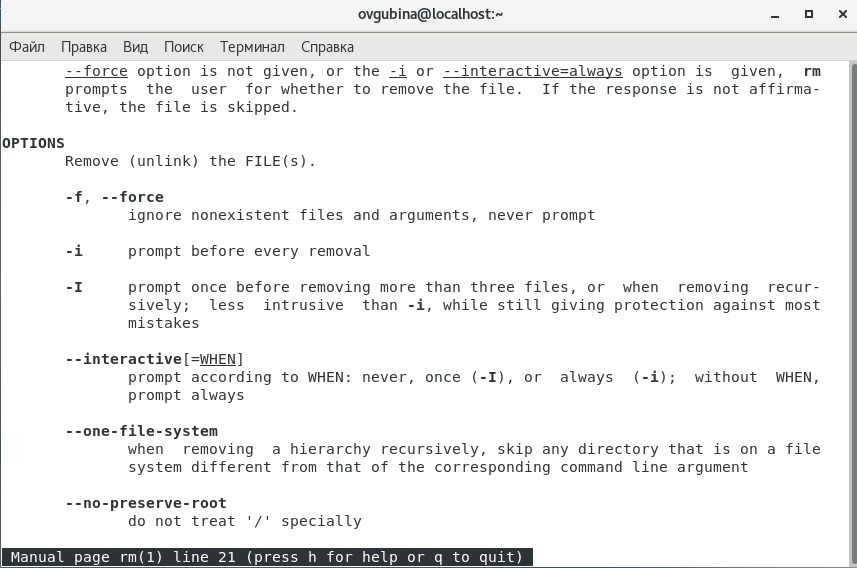
*Рисунок 24: вывод man для rmdir*



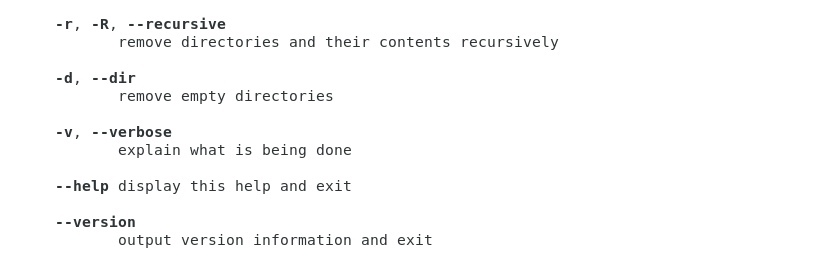
Таким образом для **rmdir** основные опции это (*рисунок 24*):

* --ignore-fail-on-non-empty - игнорирует ошибки вызванные тем, что каталог непустой;
* -p - удаляются также каталоги высшего уровня, если они пустые;
* -v - подавление сообщения, выдаваемого при действии опции -p;
* --help - вывести справочную информацию;
* --version - выводит информацию о текущей версии утилиты.

*Рисунок 25: вывод man для rm*



*Рисунок 26: вывод man для rm*



Таким образом для **rm** основные опции это (*рисунки 25-26*):

* -f - игнорировать несуществующие файлы и аргументы. Никогда не выдавать запросы на подтверждение удаления;
* i - выводить запрос на подтверждение удаления каждого файла;
* -I - выдать один запрос на подтверждение удаления всех файлов, если удаляется больше трех файлов или используется рекурсивное удаление; опция применяется, как более «щадящая» версия опции -i.
* --interactive[=WHEN] - вместо WHEN можно использовать:

never — никогда не выдавать запросы на подтверждение удаления;

once — выводить запрос один раз (аналог опции -I);

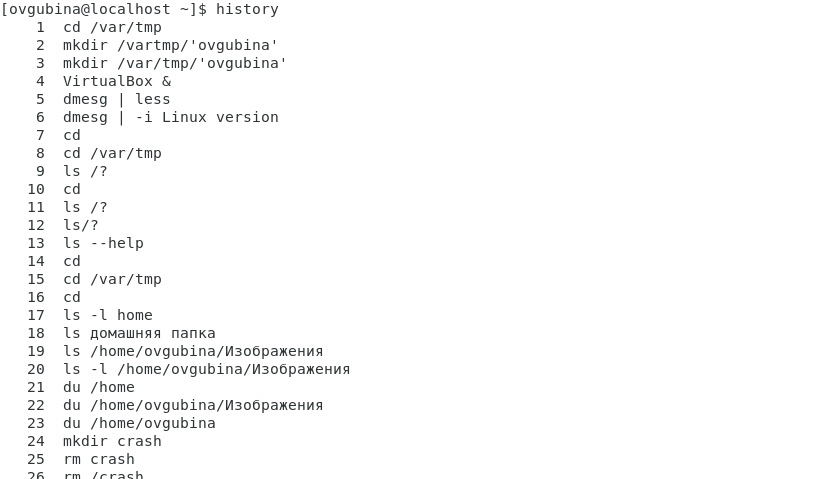
always — выводить запрос всегда (аналог опции -i);

Если значение WHEN не задано, то используется always.

* --one-file-system - во время рекурсивного удаления пропускать директории, которые находятся на других файловых системах;
* --no-preserve-root - если в качестве директории для удаления задан корневой раздел /, то считать, что это обычная директория и начать выполнять удаление;
* -r, -R, --recursive - удаление директорий и их содержимого. Рекурсивное удаление;
* -d, --dir - удалять пустые директории;
* -v, --verbose - выводить информацию об удаляемых файлах;
* --help - вывести справочную информацию;
* --version - выводит информацию о текущей версии утилиты.

1. Используя информацию, полученную при помощи команды history (*рисунок 27*), выполним модификацию и исполнение нескольких команд из буфера команд.

*Рисунок 27: вывод команды history*

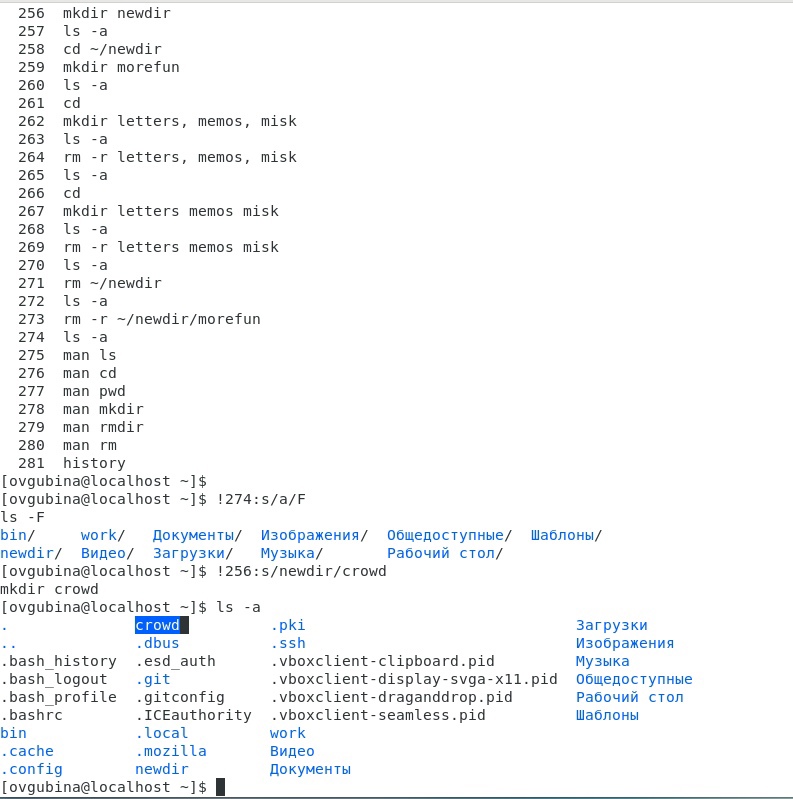


Синтаксис модификаций:

!<номер строки команды>:<s>/<что заменяем>/<на что заменяем>

Таким образом получаем следующие модификации (*рисунок 28*):

*Рисунок 28: выполнение модификаций*



После второй модификации выполнили проверку с помощью ls -a, видим, что вызов модификации прошел успешно и каталог crowd действительно был создан.

## Вывод:

Приобрела практические навыки взаимодействия пользователя с системой посредством командной строки.

### Библиография:

[1]: [Основные команды Linux](https://www.hostinger.ru/rukovodstva/osnovnyje-komandy-linux)