Отчёт по лабораторной работе №5

Дисциплина: Операционные системы

Горпинич Елена Михайловна

Содержание

[Цель работы 1](#_Toc102747381)

[Задание 1](#_Toc102747382)

[Теоретическое введение 2](#_Toc102747383)

[Команды для работы с файлами и каталогами 2](#_Toc102747384)

[Выполнение лабораторной работы 2](#_Toc102747385)

[Примеры описанные в первой части лабораторной работы 2](#_Toc102747386)

[Упражнения 4 пункт 6](#_Toc102747387)

[Вывод 10](#_Toc102747388)

[Контрольные вопросы 10](#_Toc102747389)

# Цель работы

Ознакомиться с файловой системой, её структурой, именами и содержанием каталогов. Приобрести практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

# Задание

* Выполните все примеры, приведённые в первой части описания лабораторной работы.
* Выполнить действия записанные во втором пункте лабораторной работы и зафиксировать в отчёте используемые при этом команды и результаты их выполнения
* Определить опции команды chmod, необходимые для того, чтобы присвоить перечисленным файлам выделенные права доступа, считая, что в начале таких прав нет
* Проделать приведённые в 4 пункте упражнения, записывая в отчёт по лабораторной работе используемые при этом команды
* Прочитать man по командам mount, fsck, mkfs, kill и кратко их охарактеризовать, приведя примеры.

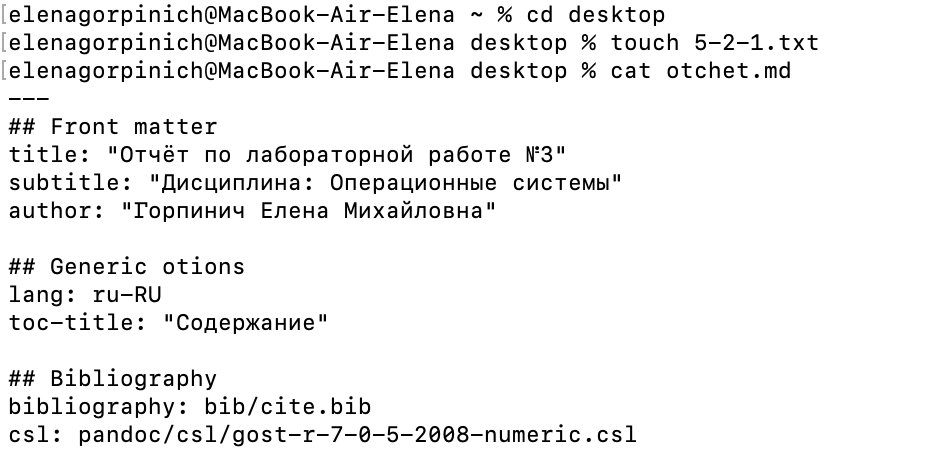
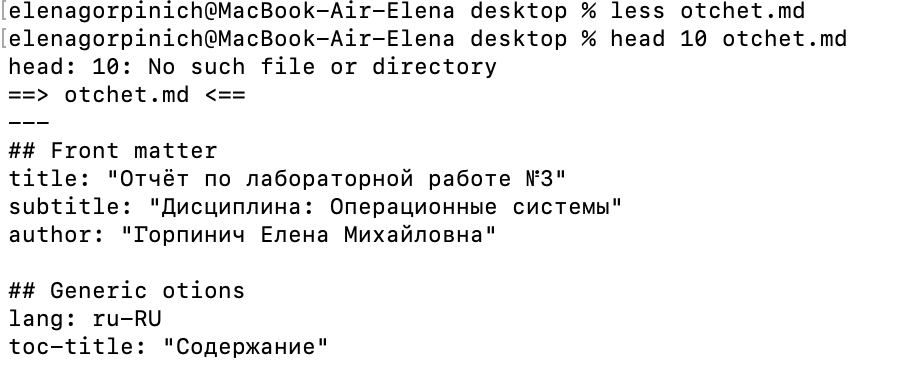
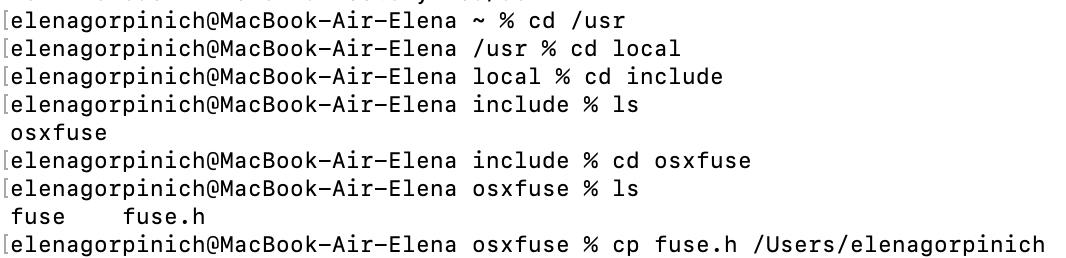
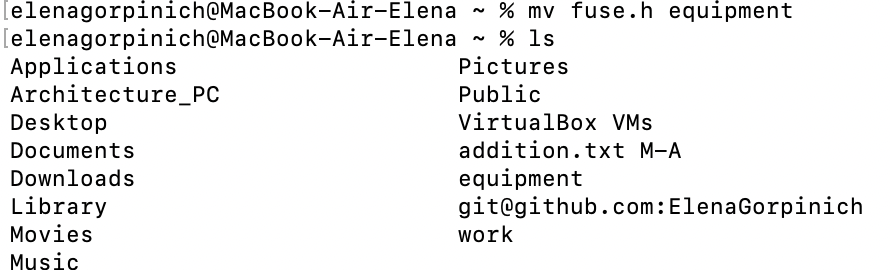
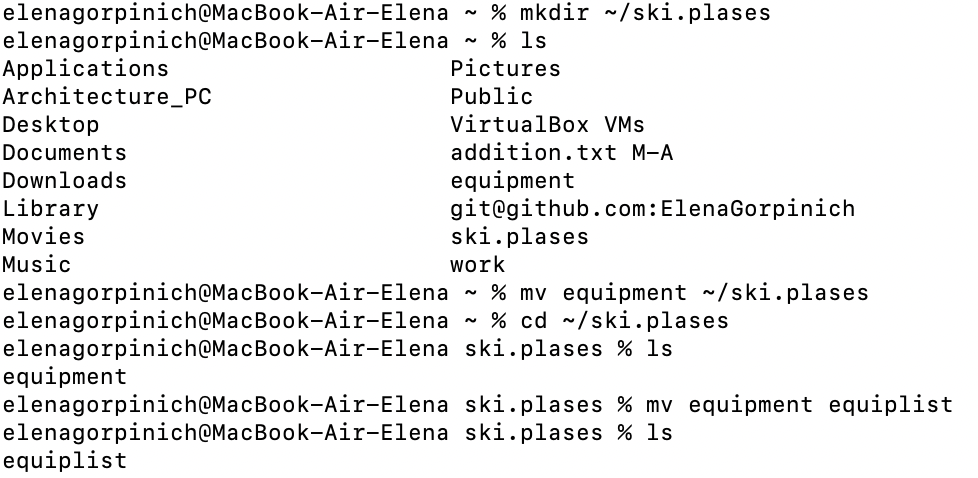
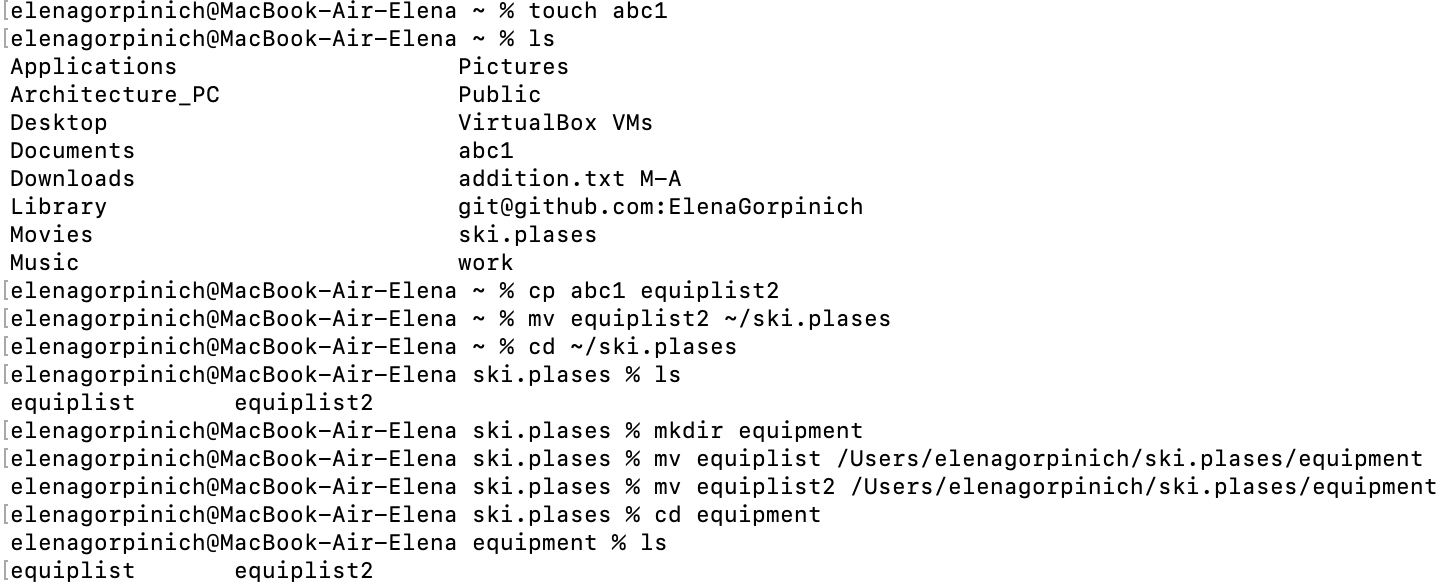
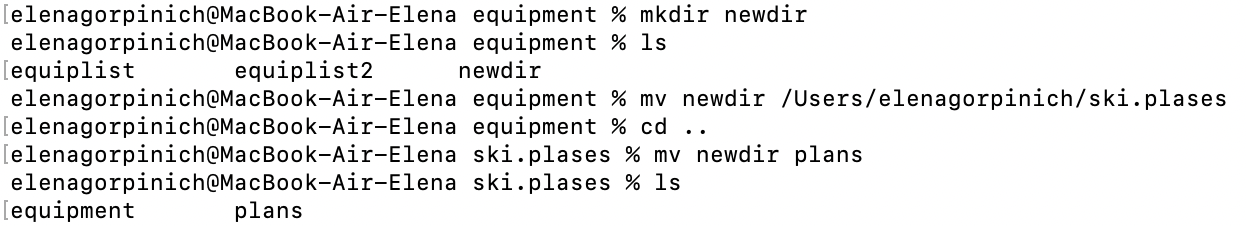
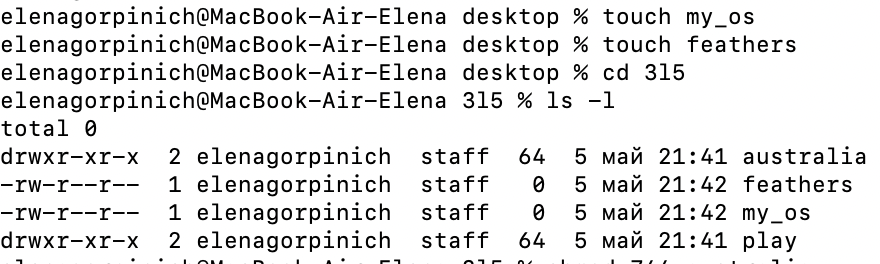
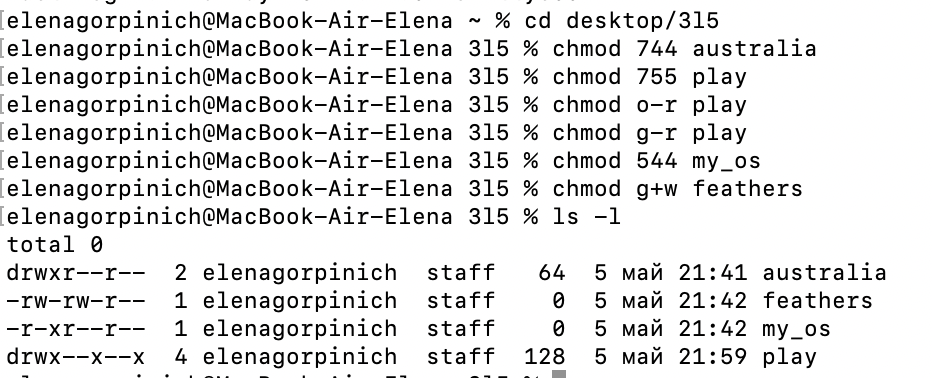
# Теоретическое введение

## Команды для работы с файлами и каталогами

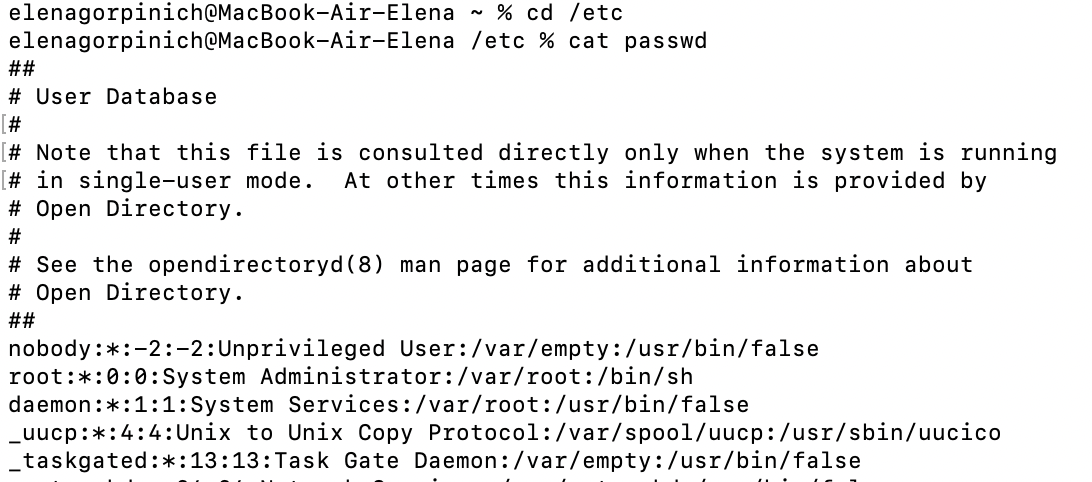
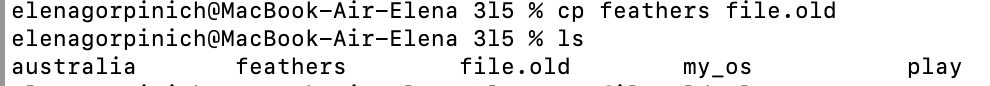
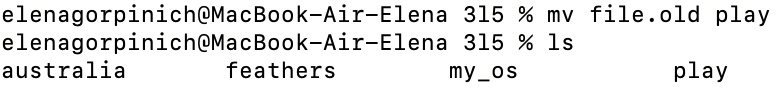
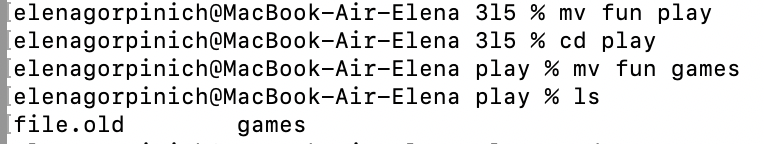
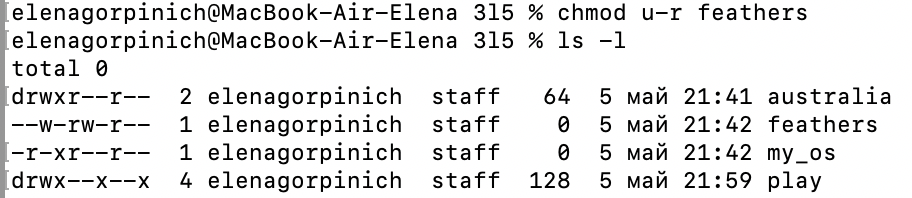
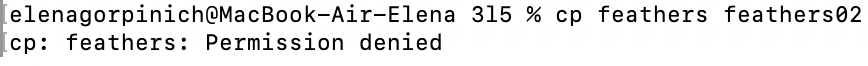
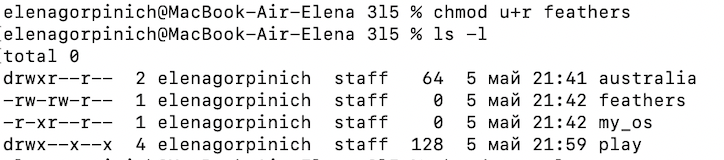
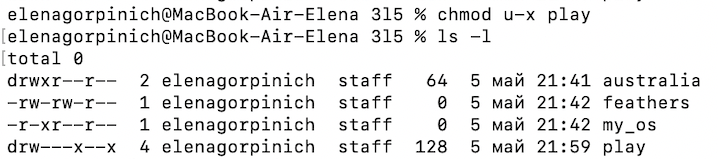
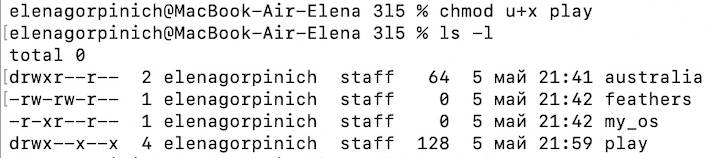
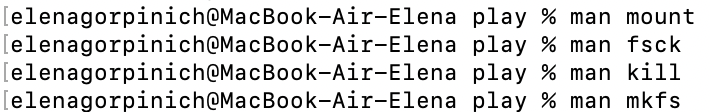
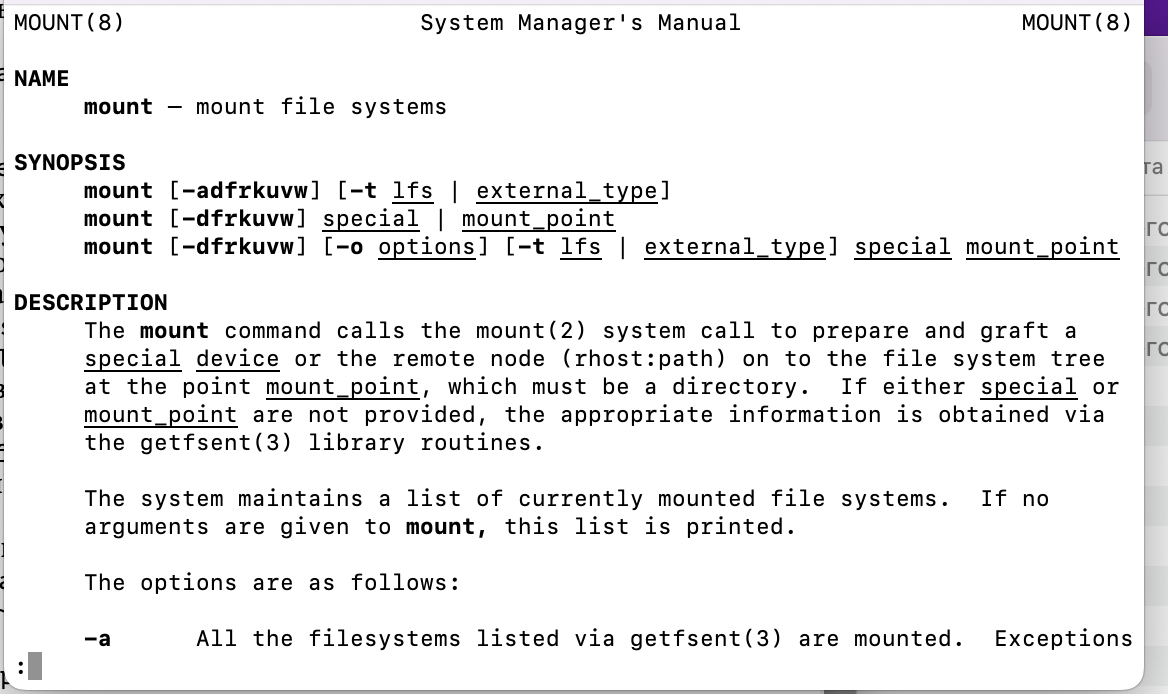
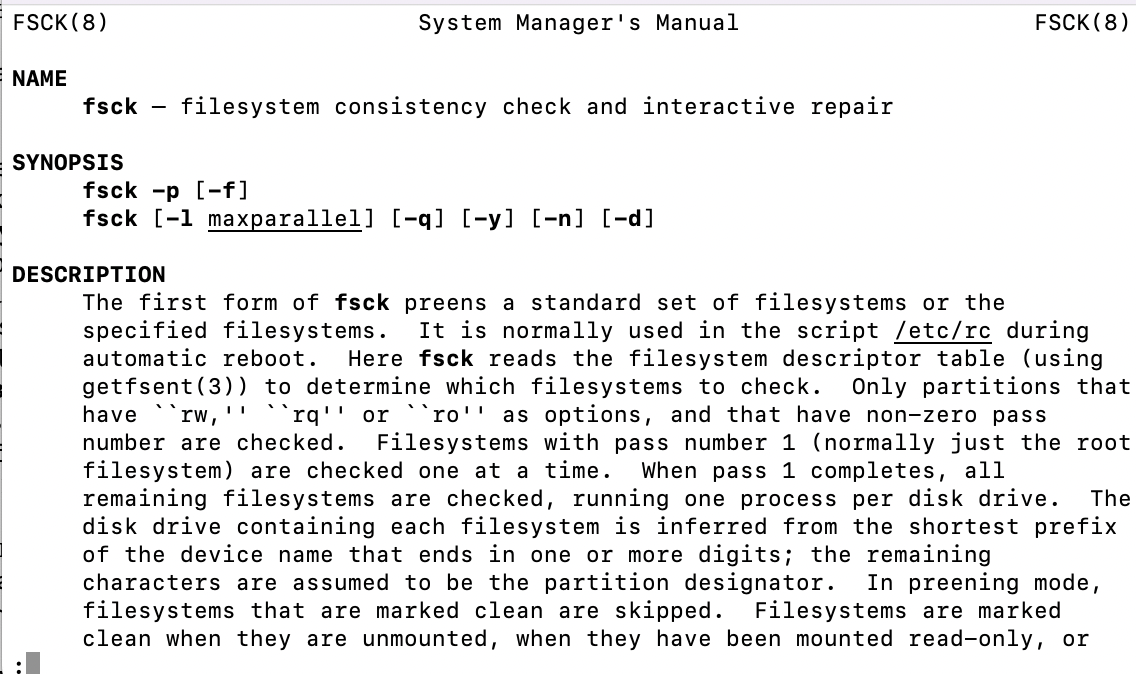
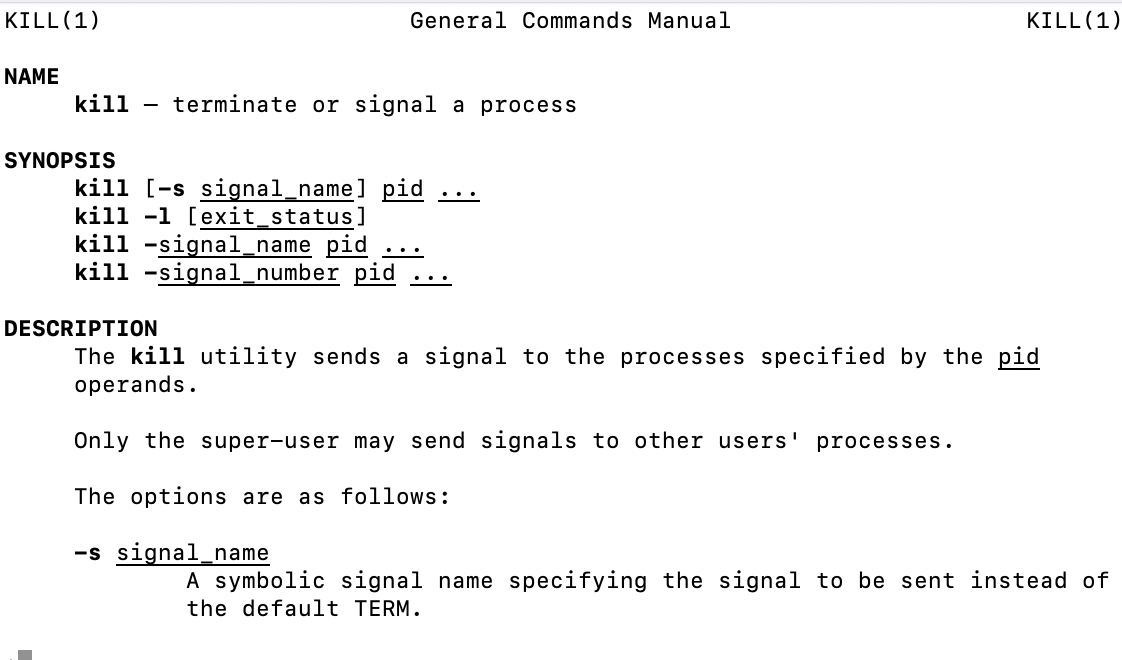
* Для создания текстового файла можно использовать команду touch.
* Для просмотра файлов небольшого размера можно использовать команду cat.
* Для просмотра файлов постранично удобнее использовать команду less.
* Команда head выводит по умолчанию первые 10 строк файла.
* Команда tail выводит умолчанию 10 последних строк файла. ## Копирование файлов и каталогов
* Команда cp используется для копирования файлов и каталогов. ## Перемещение и переименование файлов и каталогов Команды mv и mvdir предназначены для перемещения и переименования файлов и каталогов. ## Права доступа Каждый файл или каталог имеет права доступа. В сведениях о файле или каталоге указываются: – типфайла(символ(-)обозначаетфайл,асимвол(d)—каталог); – правадлявладельцафайла(r—разрешеночтение,w—разрешеназапись,x—разре- шено выполнение, - — право доступа отсутствует); – правадлячленовгруппы(r—разрешеночтение,w—разрешеназапись,x—разрешено выполнение, - — право доступа отсутствует); – правадлявсехостальных(r—разрешеночтение,w—разрешеназапись,x—разрешено выполнение, - — право доступа отсутствует). ## 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. ## Анализ файловой системы Существует несколько типов файловых систем. Перечислим наиболее часто встречаю- щиеся типы: – ext2fs(secondextendedfilesystem); – ext2fs(thirdextendedfilesystem); – ext4(fourthextendedfilesystem); – ReiserFS; – xfs; – fat(fileallocationtable); – ntfs(newtechnologyfilesystem). Для просмотра используемых в операционной системе файловых систем можно вос- пользоваться командой mount без параметров.

# Выполнение лабораторной работы

## Примеры описанные в первой части лабораторной работы

1. На рабочем столе создадим файл 5-2-1.txt, используя команду touch. Также просмотрим файл otchet.md(отчет на предыдущую лабораторную работу), с помощью команды cat(рис.[1]) 
2. При помощи команды less постранично просмотрим файл. Команда head поможет вывести на экран первые 10 строк файла (рис.[2]) 
3. Команда tail поможет вывести на экран последние 10 строк файла (рис.[3])  ## Необходимые действия пункт 2
4. Скопируем файл в домашний каталог и назовём его equipment(рис.[4-4.1])  
5. В домашнем каталоге создадим директорию ~/ski.plases.(рис.[5]). Переместим файл equipment в каталог ~/ski.plases. Переименнуем файл ~/ski.plases/equipment в ~/ski.plases/equiplist. 
6. Создайте в домашнем каталоге файл abc1 и скопируйте его в каталог ~/ski.plases, назовите его equiplist2. Создайте каталог с именем equipment в каталоге ~/ski.plases.Переместим файлы ~/ski.plases/equiplist и equiplist2 в каталог ~/ski.plases/equipment.(рис.[6]) 
7. Создадим и переместим каталог ~/newdir в каталог ~/ski.plases и назовём его plans. (рис.[7])  ## Определим опции команды chmod
8. Для начала в папке 3l5 создадим две директории - australia и play, а также два файла my\_os и feathers (рис.[8]) 
9. С помощью команды chmod присвоим созданным файлам необходимые права доступа (рис.[9]) 

## Упражнения 4 пункт

1. Просмотрим содержимое файла /etc/password.(рис.[10]) 
2. Скопируйем файл ~/feathers в файл ~/file.old. (рис.[11]) 
3. Переместим файл ~/file.old в каталог ~/play.(рис.[12]) 
4. Скопируйем каталог ~/play в каталог ~/fun.(рис.[13]) 13
5. Переместим каталог ~/fun в каталог ~/play и назовём его games.(рис.[14]) 
6. Лишим владельца файла ~/feathers права на чтение.(рис.[15]) 
7. Если мы попытаемся посмотреть файл командой cat, мы получим ошибку «отказано в разрешении» (рис.[16]) 16
8. При копировании произойдёт такая же ошибка (рис.[17]) 
9. Вернём владельцу право на чтение (рис.[18]) 
10. Лишим владельца каталога ~/play права на выполнение (рис.[19]) 
11. Из-за этого мы не сожем перейти в каталог play. Появится ошибка “permission denied”(в доступе отказано) (рис.[20]) 20
12. Вернём права владельца на выполнение каталога ~/play(рис.[21])  ## man по командам mount, fsck, mkfs, kill
13. Воспользуемся man по каждой из команд(рис.[22]) 
14. После выполнения каждой команды мы получим подробную справку по требуемым командам (рис.[23-25])    ### Краткая характеристика команд
15. mount - монтирует файловую систему Команда mount предназначена для выполнения операции монтирования файловой системы и получения информации об уже смонтированных файловых системах.

Опции -h, -V используются при вызове команды без параметров и служат для следующих целей:

* h - вывести краткую инструкцию по пользованию командой;
* V - вывести информацию о версии команды mount;

Команда mount без опций и без параметров выводит информацию обо всех уже смонтированных файловых системах. C параметрами служит для выполнения операции монтирования файловой системы.

Параметр device задает имя специального файла для устройства, содержащего файловую систему. Параметр dir задает имя точки монтирования. #### Пример - sudo mkdir /mnt/win\_d - sudo mount /dev/sda5 /mnt/win\_d

1. fsck - Filesystem Check - проверка файловой системы

В случае неправильного выключения питания ПК, при следующей загрузке, будут проверены все диски, которые не были правильно размонтированы. Данная операция может занять несколько минут. Если проверка пройдет успешно, то система загрузится нормально. #### Пример Для проверки файловой системы раздела /dev/hda1 сначала размонтируем его, а потом запустим fsck - umount /dev/hda1 - fsck /dev/hda1

1. mkfs - создаёт новую файловую систему Linux

В качестве аргумента filesys для файловой системы может выступать или название устройства или точка монтирования. Аргументом blocks указывается количество блоков, которые выделяются для использования этой файловой системой. По окончании работы mkfs возвращает 0 - в случае успеха, а 1 - при неудачной операции. В общем случае, mkfs является простым конечным интерфейсом к доступным под Linux модулям создания файловых систем, в которых вторая часть сложных имён как раз и определяет вызываемый модуль. Поиск специфического модуля создания файловой системы осуществляется примерно в следующей последовательности каталогов: /sbin, /sbin/fs, /sbin/fs.d, /etc/fs, /etc. Точный список каталогов определяется во время компиляции, но как минимум содержит /sbin и /sbin/fs, а завершается каталогами, которые перечислены в переменной окружения PATH. #### Пример Создание файловой системы типа ext2 в разделе /dev/hdb1: #mkfs -t ext2 /dev/hdb1

1. kill - посылает сигнал процессу или выводит список допустимых сигналов

Аргументы, обязательные для полных вариантов опций, являются обязательными также и для кратких вариантов.

* -s, –signal=СИГНАЛ, -СИГНАЛ имя или номер посылаемого сигнала
* -l, –list вывести имена сигналов или вывести имя сигнала, соответствующее номеру, и наоборот
* -t, –table вывести информацию о сигналах в виде таблицы
* –help вывести справку и закончить работу
* –version вывести информацию о версии и закончить работу СИГНАЛ может указываться в виде имени или номера. Также в качестве СИГНАЛА можно указывать код выхода который программа должна сообщить системе при завершении.

PID - числовой идентификатор процесса. Если число отрицательное, оно определяет группу процесса.  
#### Пример Для того чтобы вывести список всех доступных сигналов: kill -L

# Вывод

В ходе лабораторной работы я ознакомилась с файловой системой, её структурой, именами и содержанием каталогов. Мне удалось получить практические навыки по применению команд для работы с файлами и каталогами, по управлению процессами (и работами), по проверке использования диска и обслуживанию файловой системы.

# Контрольные вопросы

1. Apple File System (APFS)

Apple File System (APFS) — стандартная файловая система для компьютеров Mac с macOS 10.13 и новее, которая обеспечивает надежное шифрование, совместное использование пространства, получение моментальных снимков, быстрое изменение размеров каталогов и улучшенные принципы файловой системы. Система APFS оптимизирована для флеш-накопителей и дисков SSD, которые используются в новейших компьютерах Mac, однако ее также можно использовать для более старых систем с традиционными жесткими дисками и внешними накопителями с прямым подключением. Версии macOS 10.13 и новее поддерживают APFS для загрузочных томов и томов данных. APFS выделяет дисковое пространство внутри контейнера (раздела) по требованию. Если контейнер APFS содержит несколько томов, его свободное пространство используется совместно и при необходимости автоматически назначается любому из томов. При желании можно задать размеры резерва и квоты для каждого тома. Каждый том использует только часть общего контейнера, поэтому доступное пространство равно общему размеру контейнера за вычетом размера всех томов в контейнере.

1. mount - монтирует файловую систему
2. Отсутствие синхронизации между образом файловой системы в памяти и ее данными на диске в случае аварийного останова может привести к появлению следующих ошибок:

* Один блок адресуется несколькими mode (принадлежит нескольким файлам).
* Блок помечен как свободный, но в то же время занят (на него ссылается onode).
* Блок помечен как занятый, но в то же время свободен (ни один inode на него не ссылается).
* Неправильное число ссылок в inode (недостаток или избыток ссылающихся записей в каталогах).
* Несовпадение между размером файла и суммарным размером адресуемых inode блоков.
* Недопустимые адресуемые блоки (например, расположенные за пределами файловой системы).
* “Потерянные” файлы (правильные inode, на которые не ссылаются записи каталогов).
* Недопустимые или неразмещенные номера inode в записях каталогов.

1. mkfs - позволяет создать файловую систему
2. Cat - выводит содержимое файла на стандартное устройство вывода
3. Cp – копирует или перемещает директорию, файлы
4. Mv - переименовать или переместить файл или директорию
5. Права доступа к файлу или каталогу можно изменить, воспользовавшись командой chmod. Сделать это может владелец файла (или каталога) или пользователь с правами администратора.