Отчёт по лабораторной работе №4

Основы интерфейса взаимодействия пользователя с системой Unix на уровне командной строки

Тагиев Павел Фаикович

Содержание

# 1 Цель работы

Приобретение практических навыков взаимодействия пользователя с системой посредством командной строки.

# 2 Задание

1. Определить полное имя домашнего каталога.
2. Выполнить следующие действия:
   1. Перейти в каталог /tmp.
   2. Вывести на экран содержимое каталога /tmp командой ls с различными опциями.  
      Пояснить разницу выводимой информации.
   3. Определить есть ли в каталоге /var/spool/ подкаталог с именем cron.
   4. Перейти в домашний каталог и вывести его содержимое. Определить кто является владельцом файлов и подкаталогов.
3. Выпольнить следующие действия:
   1. В домашнем каталоге создать новый каталог с именем newdir.
   2. В каталоге ~/newdir создать новый каталог с именем morefun.
   3. Создать одной командой в домашнем каталоге три новый каталога с именами  
      letters,memos,misk. Затем удалить эти каталоги одной командой.
   4. Попробовать удалить ранее созданный каталог ~/newdir командой rm. Проверить был ли удален каталог.
   5. Удалить каталог ~/newdir/morefun из домашнего каталога. Проверит был ли удален каталог.
4. С помощью команды man определить, какую опцию команды ls нужно использовать для просмотра не только указаного каталога, но  
   и его подкаталогов.
5. С помощью команды man определить набор опций команды ls, позволяющий отсортировать по времени последнего изменения выводимый список  
   содержимого каталога развернутым описанием файлов.
6. Использовать команду man для просмотра описания следующих команд: cd, pwd, mkdir, rmdir, rm.  
   Поясните основные опции этих команд.
7. Используя ифнормацию полученную при помощи команду history, выполните модификацию и исполнение нескольких команд из буфера команд.

Задание сформировано в соответствии с [1].

# 3 Теоретическое введение

В операционной системе типа Linux взаимодействие пользователя с системой обычно осуществляется с помощью командной строки посредством построчного ввода команд. При этом обычно используются командные интерпритаторы языка shell. Такие как /bin/sh; /bin/csh; /bin/ksh /bin/zsh, мы же используем /bin/bash.

Общий формат команд можно представить следующим образом:

<имя\_команды><разделитель><аргументы>

Далее в отчете мы подробно рассмотрим и испольуем команды:  
cd, pwd, ls, mkdir, rmdir, rm и man.

# 4 Выполнение лабораторной работы

## 4.1 Некоторые необходимые сведения

Перед основной частью отчета, нужно уточнить некторые моменты. Эту работу я выполнял на WSL с установленной ОС Ubuntu. В терминале я использую немного измененный prompt(рис. 1). Если смотреть слева направо, то сначала идет имя\_пользователя@имя\_хоста потом в квадратных скобках текущая директория: [текущая\_директория]. Затем если мы находимся в git репозитории отображается [имя\_ветки коммит]. В конце я добавил нумерацию prompt, которая сбрасывается когда вызывается команда clear или нажимается комбинация клавишь Ctrl + l. Так мне будет удобнее ссылаться на части терминала.

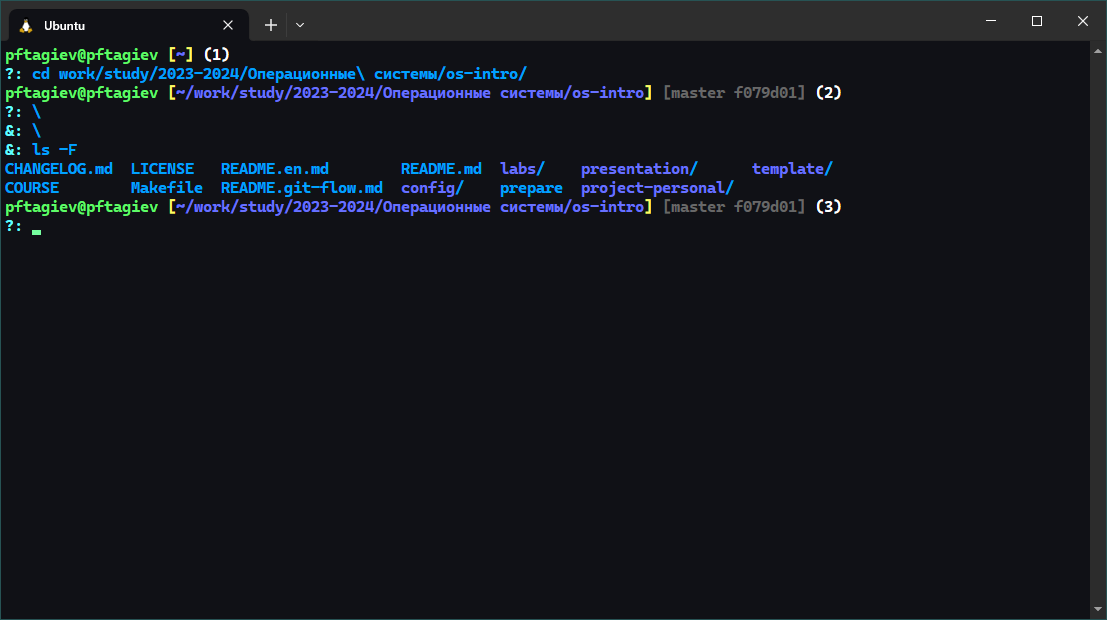


Рис. 1: Мой prompt

Ввод команд всегда начинается с новой строки после ?:. В остальном, это обычная ОС Ubuntu с командной оболочкой bash. Подробнее о настройке prompt можно узнать в [2]. Код меняющий prompt расположен в конце раздела на лист. 1 его нужно поместить в файл ~/.bashrc. Теперь мы можем перейти к основной части отчета.

Листинг 1: Код изменяющий prompt

parse\_git\_info() {  
 branch=$(git rev-parse --abbrev-ref HEAD 2> /dev/null)  
 if [ -z "$branch" ]; then  
 return  
 fi  
  
 commit=$(git rev-parse --short HEAD 2> /dev/null)  
 if [ -n "$commit" ]; then  
 commit=" $commit"  
 fi  
  
 echo -e "\e[90m[$branch$commit] "  
}  
  
PORMPT\_NUMBER=1  
  
clear() {  
 $(which clear)  
 PORMPT\_NUMBER=1  
}  
  
update\_prompt() {  
 PS1="\\[\e[1;92m\\]\u@\H\\[\e[1;33m\\] "  
 PS1+="[\\[\e[1;94m\\]\w\\[\e[1;33m\\]] "  
  
 PS1+="$(parse\_git\_info)"  
  
 PS1+="\\[\e[37m\\]($PORMPT\_NUMBER) "  
  
 PS1+="\n"  
 PS1+="\\[\e[1;96m\\]?: "  
 PS1+="\\[\e[0m\\]"  
  
 ((++PORMPT\_NUMBER))  
}  
  
PROMPT\_COMMAND="update\_prompt"  
  
PS2="\\[\e[1;96m\\]&: \\[\e[0m\\]"

## 4.2 Полное имя домашней директории

Чтобы узнать полное имя домашней директории текущего пользователя можно воспользоваться командой pwd вызвав ее из ~/. Или же можно вывести на экран значение константы HOME из любой директории системы. Пример можно увидеть на рис. 2.

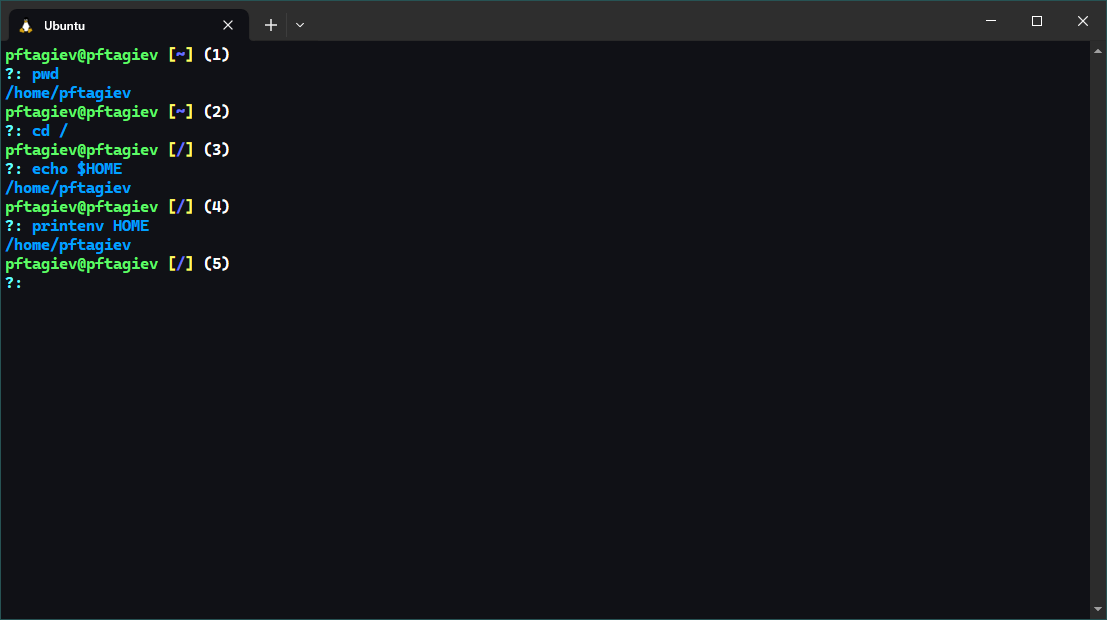


Рис. 2: Полное имя домашней директории

## 4.3 Директория /tmp и ее содержимое

В директории /tmp хранятся временные файлы которые необходимы для запущенного программного обеспечения. Перейдем в нее командой cd как показано на рис. 3. Используем команду ls, мы видем 4 временных файла. Узнаем о них чуть больше добавив флаг -l как показано в (3). Как видно они были созданы 5 апреля от имени пользователя root и только у него есть права на чтение, редактирование и исполнение этих файлов.

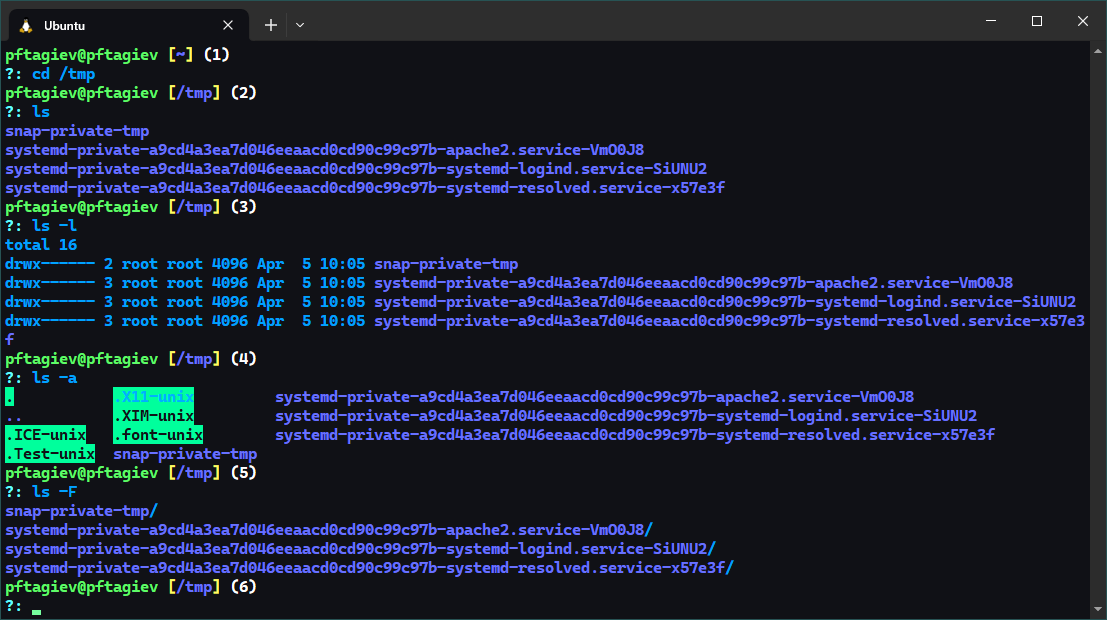


Рис. 3: Директория tmp

Посмотрим есть ли в этой папке скрытые файлы, добавив опцию -a. Как видно на рис. 3 они есть. Теперь, наконец, узнаем тип этих файлов добавив флаг -F, в соответсвии с табл. 1 это каталоги.

Таблица 1: Символы определяющие тип файла

| Тип файла | Символ |
| --- | --- |
| Каталог | / |
| Исполняемый файл | \* |
| Ссылка | @ |

## 4.4 Файл cron

Перейдем в каталог /var/spool/ как того требует задание. И с помощью комбинации команд ls и grep узнаем есть ли в этом каталоге файл cron. Результат можно увидеть на рис. 4

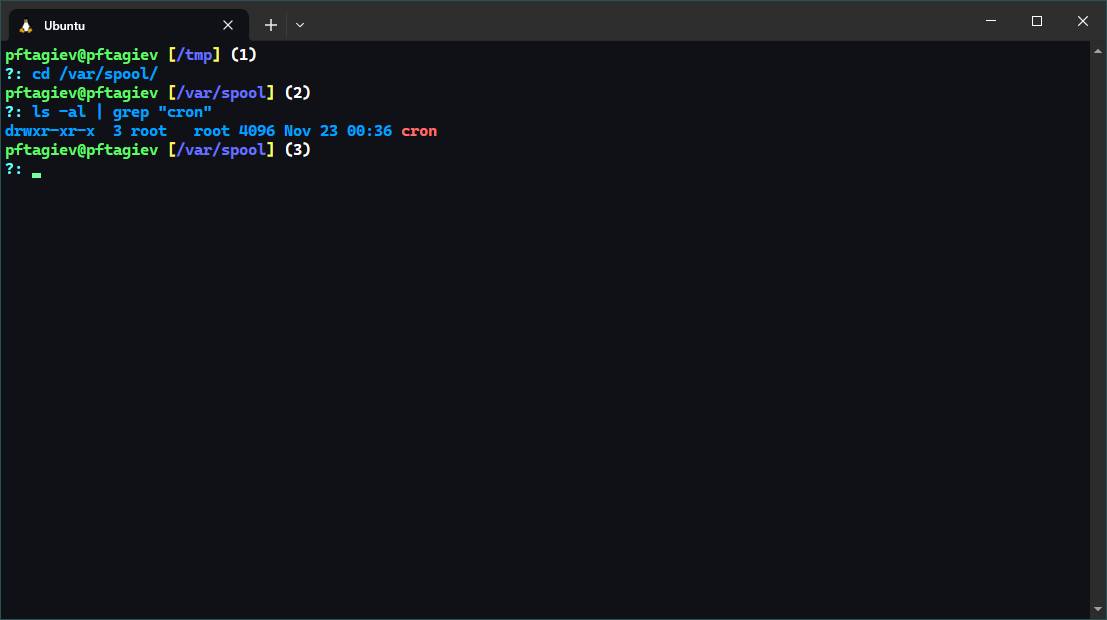


Рис. 4: Каталог /var/spool/

## 4.5 Домашний каталог

Вернемся в домашний каталог используя cd без аргументов. Выведем его содержимое командой ls -alF. Как видно на рис. 5 всеми файлами этой директории владеет пользователь pftagiev, а директорией выше уже пользователь root.

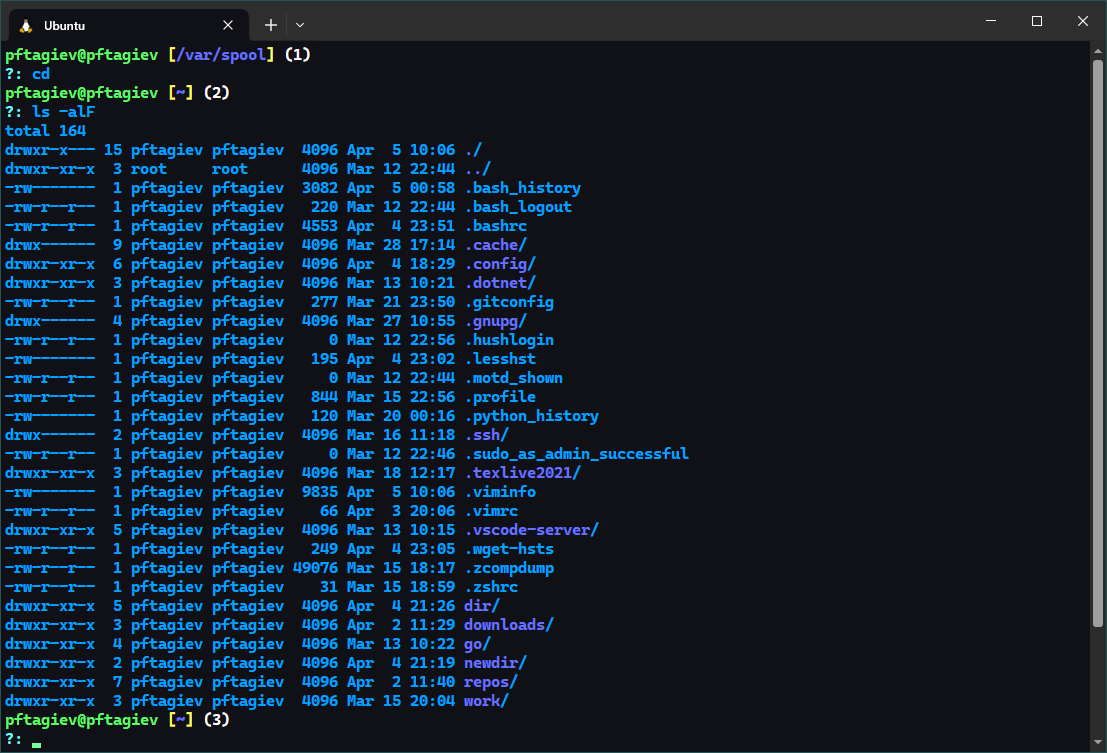


Рис. 5: Домашний каталог

## 4.6 Использование mkdir, rmdir и rm

Разберем что происходит на рис. 6, с помощью списка где каждый пункт соответствует prompt на изображении.

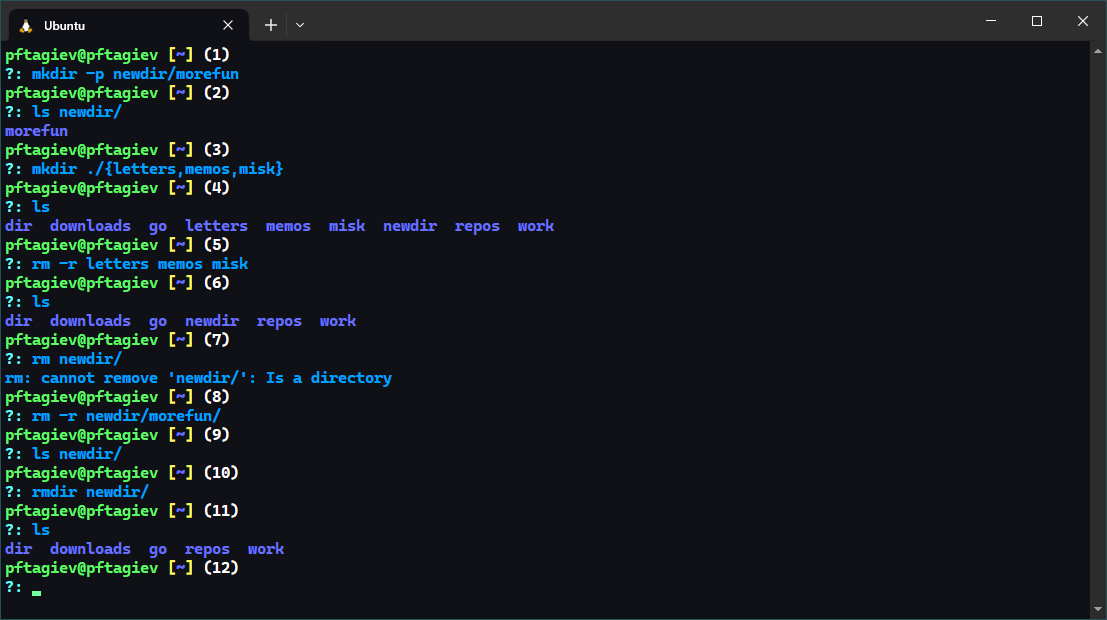


Рис. 6: Создание и удаление каталогов

1. Создадим каталог ~/newdir/morefun командой mkdir с ключем -p, который означает создать все родительские  
   каталоги для конечного каталога.
2. Проверим что конеченый каталог был создан.
3. Создадим одной командой каталоги letters, memos, misk.
4. Проиверим их наличие.
5. Удалим каталоги letters, memos, misk одной командой.
6. Проверим что каталоги были удалены.
7. Попробуем удалить ранее созданный каталог ~/newdir командой rm без флагов. Неудача, система сообщает что не может удалить ~/newdir так как это каталог.
8. Удалим каталог ~/newdir/morefun командой rm -r для рекурсивного удаления.
9. Проверим что ~/newdir/morefun был удален.
10. Удалим ~/newdir/ командой rmdir.
11. Проверим что каталог был удален.

## 4.7 Рекурсивный вывод команды ls

Требуется найти в документации к команде ls флаг для вывода не только указанного каталога но и всех его подкаталогов. Введем в терминале команду man ls как показано на рис. 7.

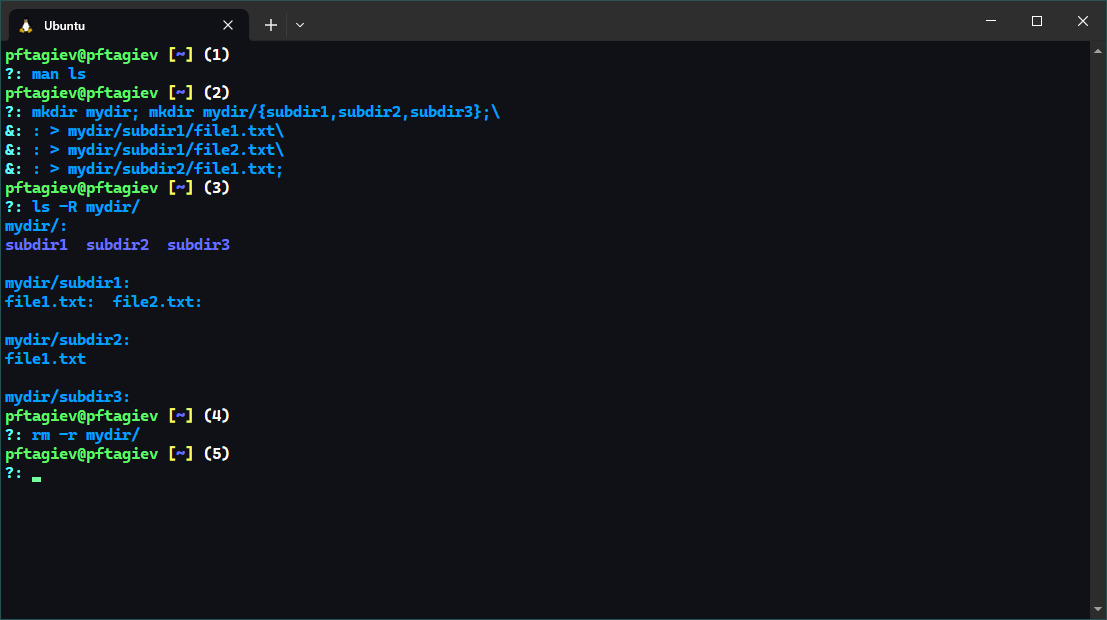


Рис. 7: Рекурсивный вывод

Откроется документация к команде ls. Воспользуемся поиском по документации введя /то\_что\_ищем. Как видно на рис. 8 нам нужна опция -R. На рис. 7 в (3) можно увидеть использование этой опции.

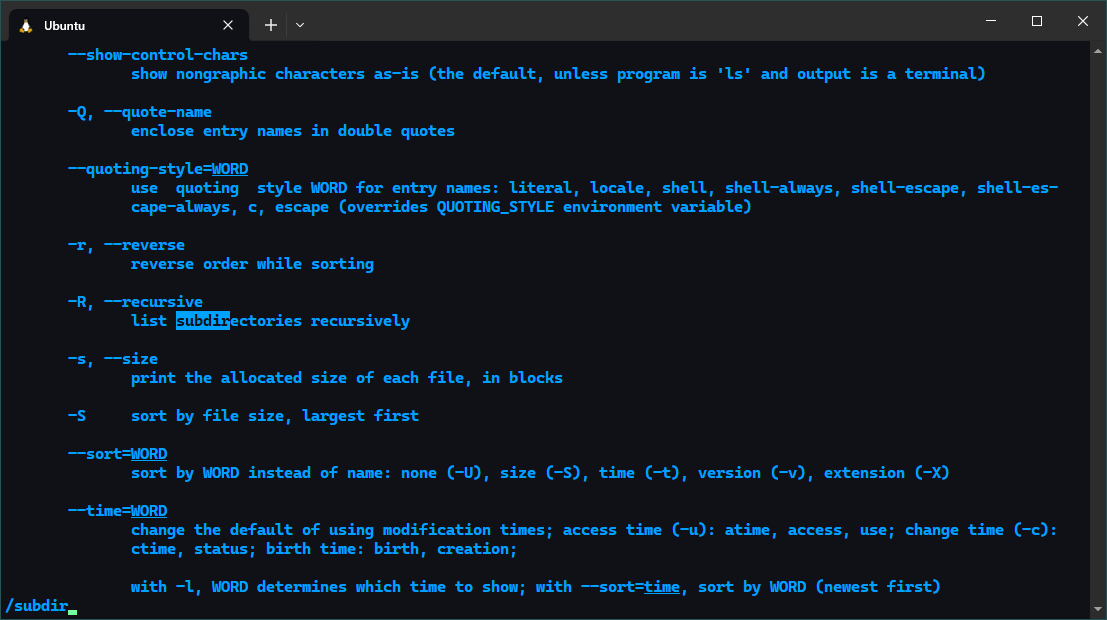


Рис. 8: Документация ls

## 4.8 Сортировка вывода ls по времени последнего изменения

Снова воспользуемся документацией к команде ls, но теперь введем в поиске фразу sort by time (рис. 9). Исходя из документации, чтобы отсортировать вывод ls по времени последнего изменения нужно использовать ключ -t, пример использования этого ключа можно увидеть на рис. 10.

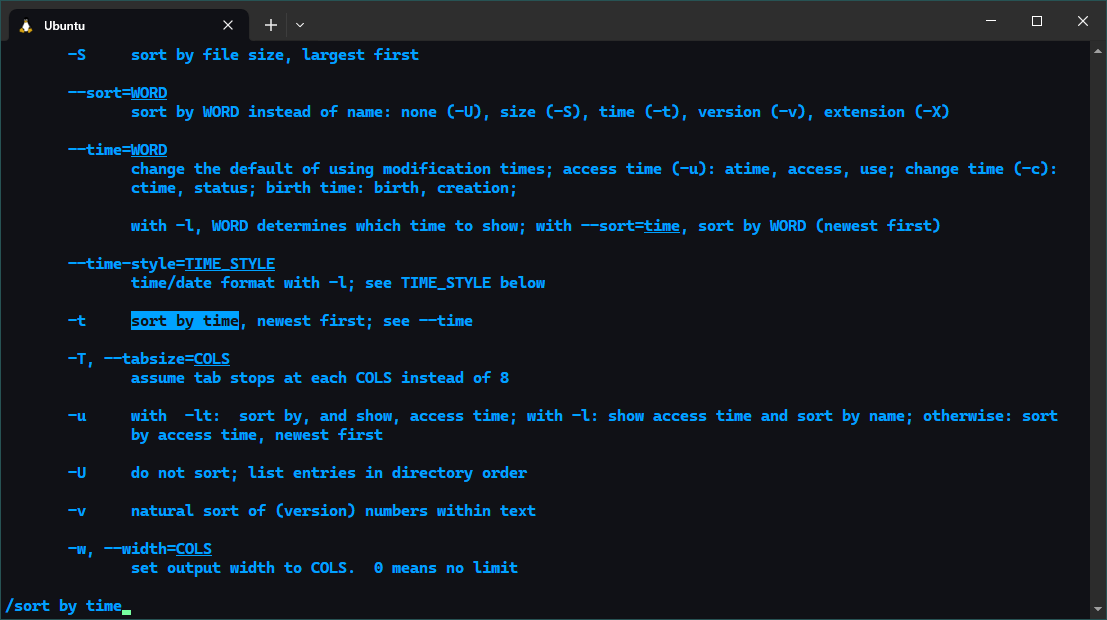


Рис. 9: Документация ls, sort by time

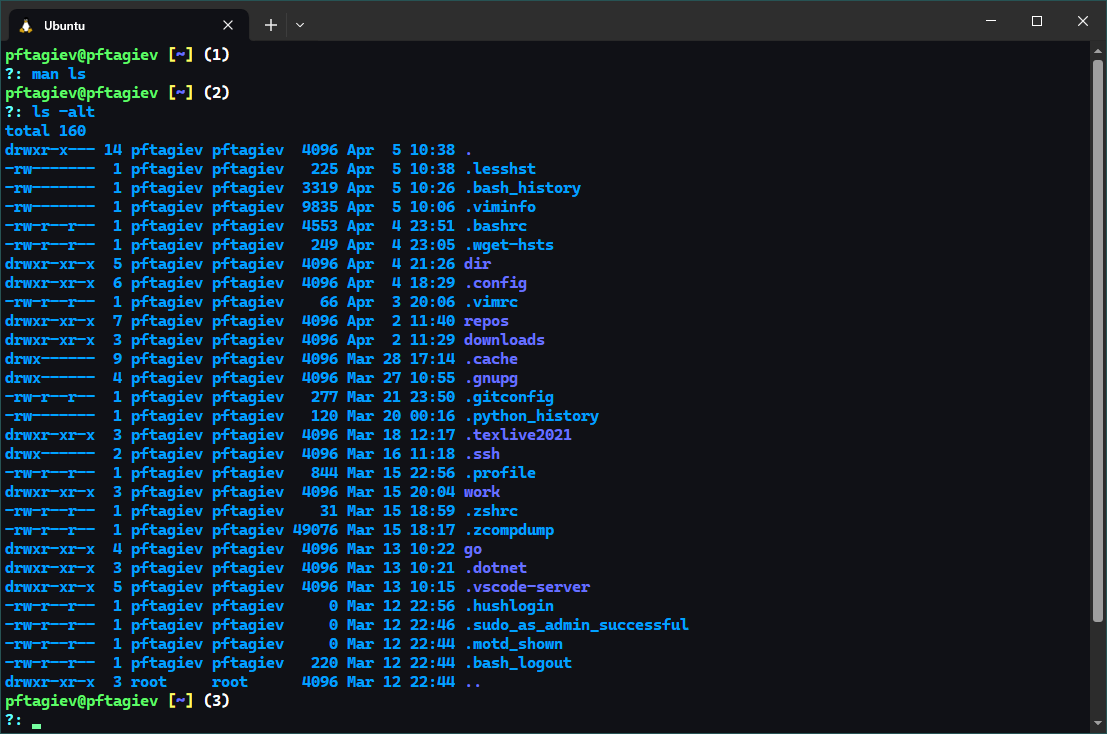


Рис. 10: Сортировка вывода ls

## 4.9 Документация к команде cd

К сожелению, так как команда cd является встроенной для нее нельзя получить документацию через утилиту man (рис. 11 pormpt (1) и (2)). Но можно получить короткую справку через help. Чуть позже мы разберем флаги cd, которые вывела в терминал команда help, но прежде нужно понять что такое symbolic link.

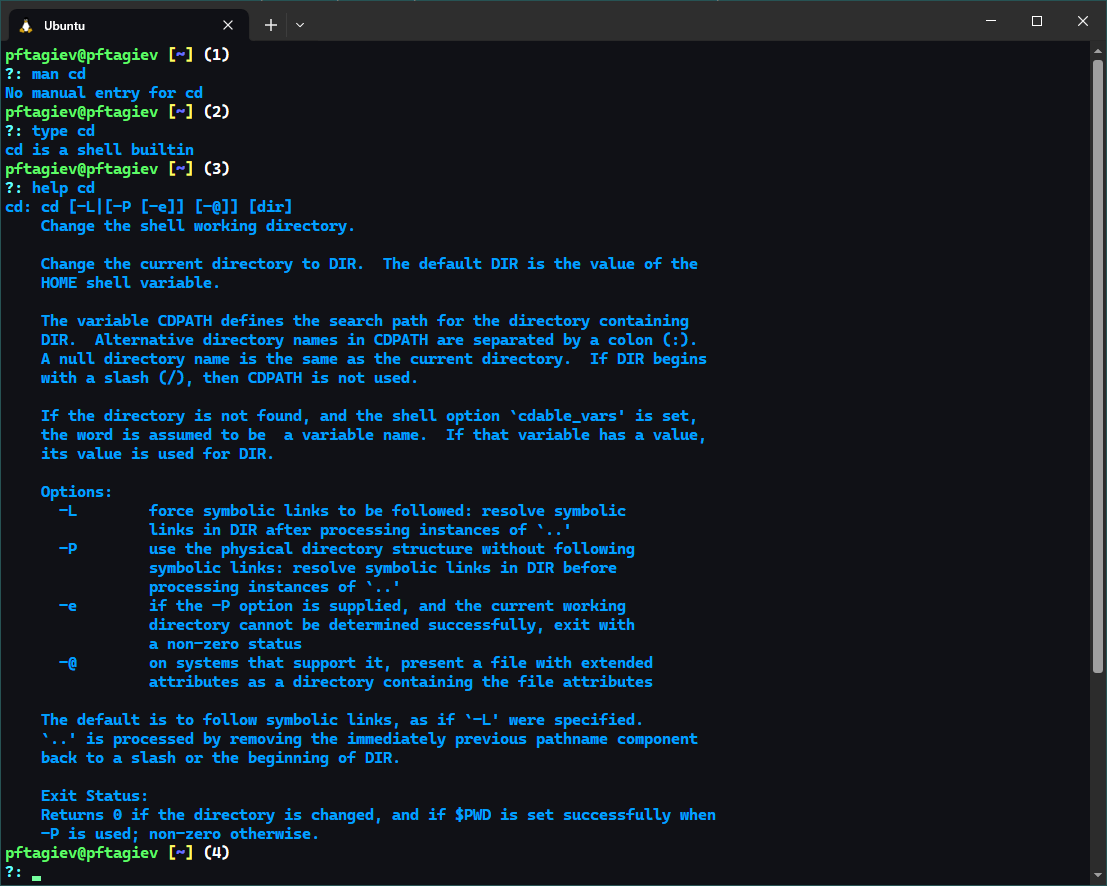


Рис. 11: Команда help для cd

Если проводить аналогию с Windows, то ***symlink*** или символические ссылки, это не что иное, как ярлыки. В Linux ссылки бывают двух видов ***hard*** и ***soft*** в табл. 2 можно увидеть их различия.

Таблица 2: Различия ссылок

| SoftLink | HardLink |
| --- | --- |
| Ссылается как на файл так и на каталог | Ссылается только на файл |
| Просто следуют по пути до файла или каталога | Указывает на фактические данные файла в томе хранения |
| Указывает на файлы и каталоги как на локальных так и на внешних томах | Может указывать на файлы только в локальном томе |
| Не работает если файл или каталог были удалены | Работает после удаления файла |

***Hard*** ссылки создаются командой ln <имя\_таргета> <имя\_ссылки>, ***soft*** ссылки создаются той же командой, но с флагом -s. Подробнее разобраться с ссылками можно, например, в [3].

Вернемся к команде cd и ее флагам (рис. 11).

* -L - Переходить по символическим ссылкам. Данное поведение используется по умолчанию.
* -P - Разыменовывать символические ссылки. В данном случае, если осуществляется переход на символическую ссылку, которая указывает на директорию, то в результате команда cd изменит текущую рабочую директорию на директорию, указанную в качестве параметра
* -P -e - Выйти с ошибкой, если директория, в которую осуществляется переход, не найдена.

## 4.10 Документация к команде pwd

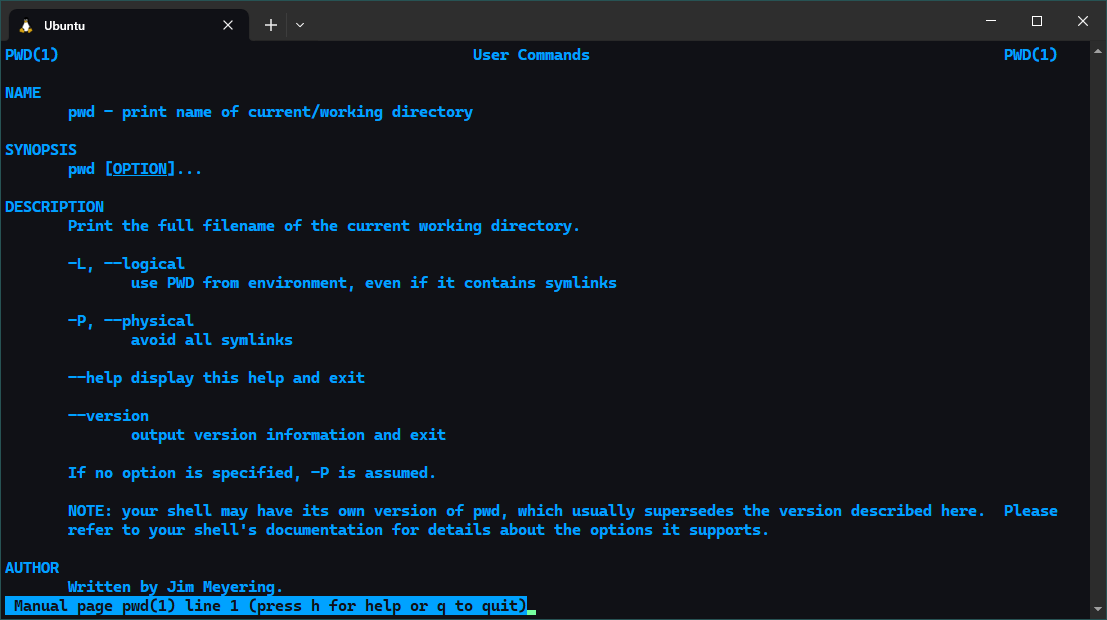


Рис. 12: Документация pwd

Команда pwd выводит в терминал текущую рабочую директорию. Разберем ее основные флаги, котороые показаны на рис. 12.

* -L - Не разыменовывать символические ссылки. Если путь содержит символические ссылки, то выводить их без преобразования в исходный путь.
* -P - Преобразовывать символические ссылки в исходные имена. Если путь содержит символические ссылки, то они будут преобразованы в названия исходных директорий, на которые они указывают.

## 4.11 Документация к команде mkdir

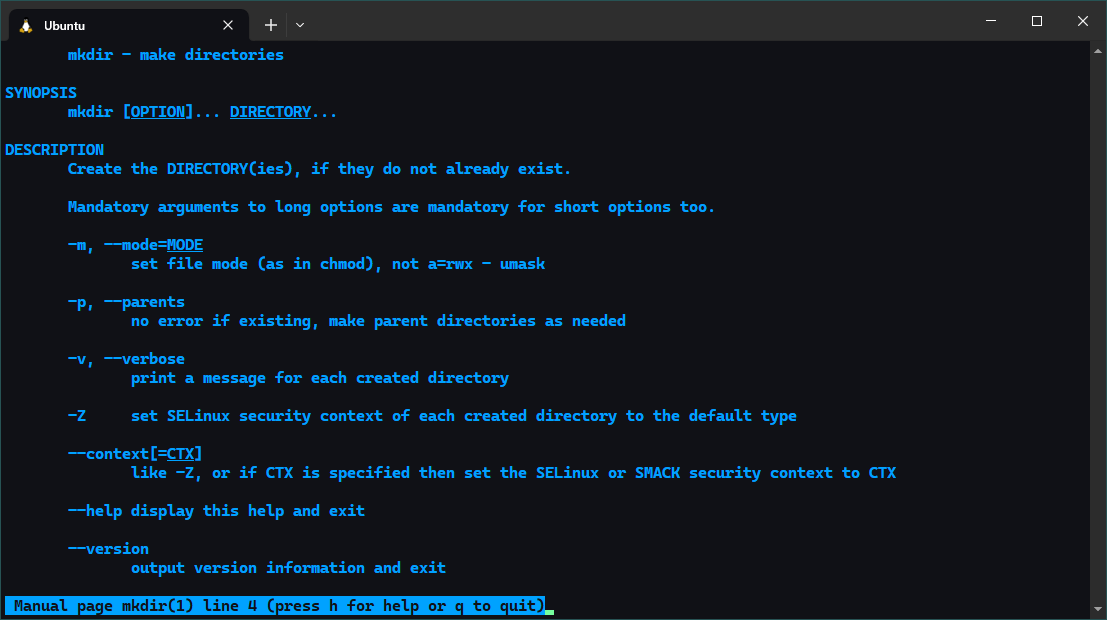


Рис. 13: Документация mkdir

mkdir - создает каталог. Ее флаги можно увидеть на рис. 13, их перевод ниже в виде списка.

* -m - Устанавливает права доступа для создаваемой директории.
* -p - Создать все директории, которые указаны внутри пути.  
  Если какая-либо директория существует, то предупреждение об этом не выводится.
* -v - Выводить сообщение о каждой создаваемой директории.
* -Z - Установить контекст SELinux для создаваемой директории по умолчанию.
* --context[-CTX] - Установить контекст SELinux для создаваемой директории в значение CTX.

## 4.12 Документация к команде rmdir

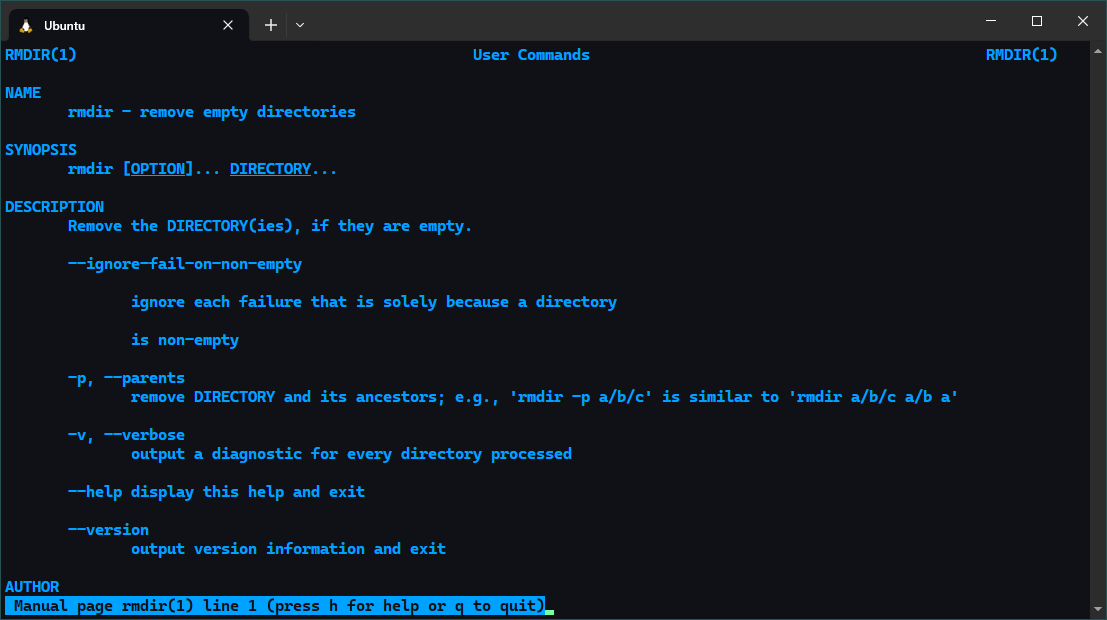


Рис. 14: Документация rmdir

rmdir - удаляет пустые каталоги (рис. 14).

* --ignore-fail-on-non-empty - Игнорировать неудачное удаление непустых каталогов.
* -p - Удалить директорию и ее родительские директории.
* -v - Вывод информации по каждой удаленной директории.

## 4.13 Документация к команде rm

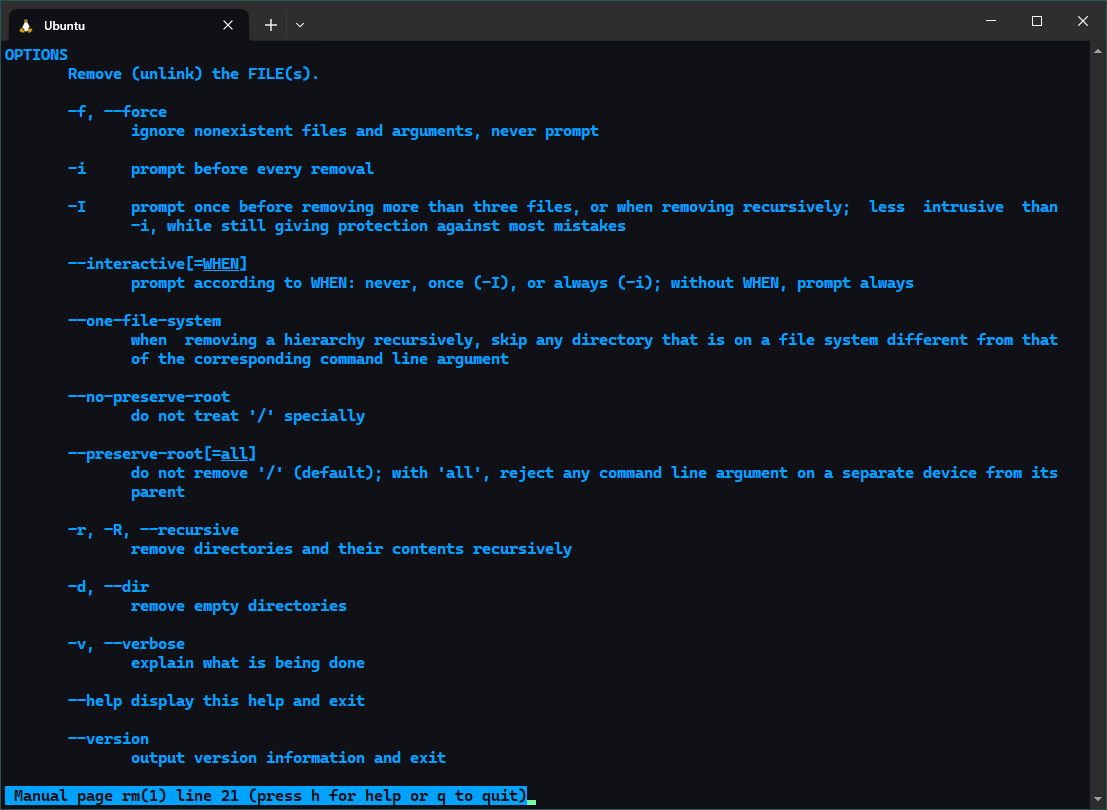


Рис. 15: Документация rm

Команда rm - удаляет файлы и каталоги(если применить специальный флаг). Документация полученная командой man на рис. 15, ниже перевод.

* -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.
* --interactive[never|once|always] - аналог -i и -I.
* --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.
* --preserve-root - Если в качестве директории для удаления задан корневой раздел /,  
  то запретить выполнять команду rm над корневым разделом. Данное поведение используется по умолчанию.
* -r - Удаление директорий и их содержимого. Рекурсивное удаление.
* -d - Удалять пустые директории.
* -v - Выводить информацию об удаляемых файлах.

## 4.14 Модификация команд из history

Выведем на экран историю команд используя history, как показано на рис. 16.

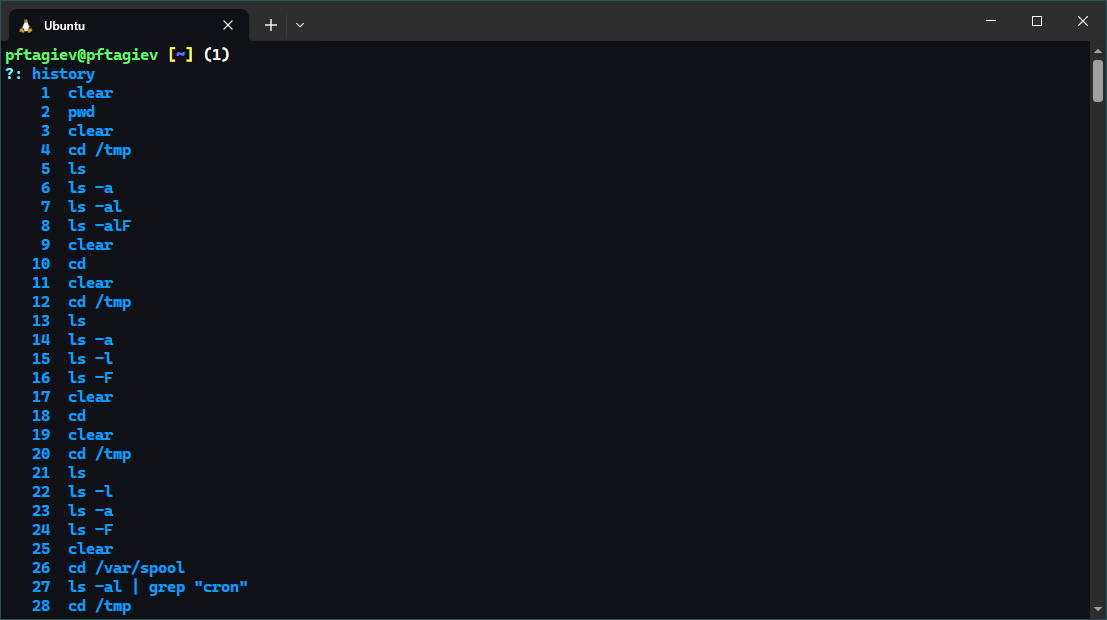


Рис. 16: Использование history

Вызовем модифицированные команды из буфера команд как показано на рис. 17 в prompt (2) - (4).

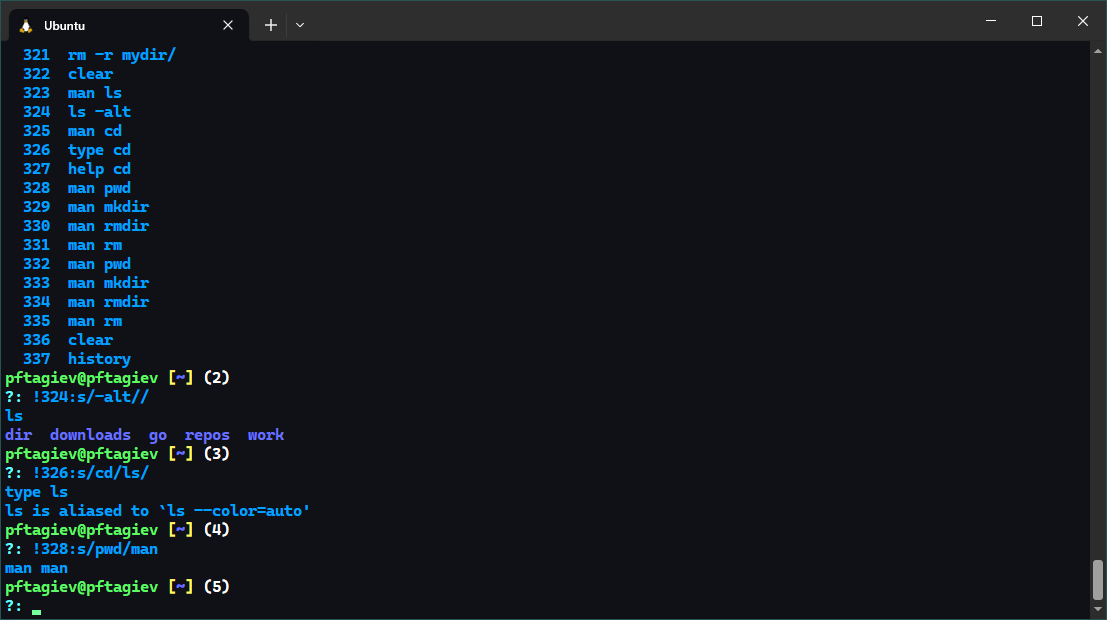


Рис. 17: Использование модифицированных команд

# 5 Ответы на контрольные вопросы

1. Что такое командная строка?  
   Это способ взаимодействия между человеком и компьютером путём отправки компьютеру команд, представляющих собой последовательность символов. Команды интерпретируются с помощью специального интерпретатора, называемого оболочкой. Интерфейс командной строки противопоставляется системам управления программой на основе меню, а также различным реализациям графического интерфейса [4].
2. При помощи какой команды можно определить абсолютный путь текущего каталога? Приведите пример.  
   Это можно сделать при помощи команды pwd. Например, находясь в каталоге ~/downloads, команда pwd выведет следующее  
   /home/имя\_пользователя/downloads.
3. При помощи какой команды и каких опций можно определить только тип файлов и их имена в текущем каталоге? Приведите примеры.  
   При помощи команды ls с флагом -F, пример можно увидеть на рис. 3 в prompt (5).
4. Каким образом отобразить информацию о скрытых файлах? Приведите примеры.  
   Для этого нужно испльзовать команду ls с флагом -a, пример уже встречался в отчете на рис. 3 в prompt (4).
5. При помощи каких команд можно удалить файл и каталог? Можно ли это сделать одной и той же командой? Приведите примеры.  
   Файл можно удалить командой rm, каталог, если он пустой, удаляется командой rmdir. Командой rm можно удалить и каталог, но нужно добавить флаг -r.
6. Каким образом можно вывести информацию о последних выполненных пользователем командах?  
   Можно использовать history N, где N - количество последних команд.
7. Как воспользоваться историей команд для их модифицированного выполнения? Приведите пример.  
   Нужно использовать конструкцию: !номер\_команды:s/паттерн/на\_что\_меняем/. Номер команды можно получить при помощи команды history. Пример использования можно увидеть на рис. 17.
8. Приведите примеры запуска нескольких команд в одной строке.
   * mkdir my\_dir; cd my\_dir; touch file.txt
   * cd; ls -alF
   * touch file.txt; mv file.txt ~
9. Дайте определение и приведите примеры символов экранирования.  
   Экранирование символов - замена в тексте управляющих символов на соответствующие им последовательности символов[5].
   * mkdir my\ dir\ 1 - экранирование пробела при создании директории.
   * echo \"text\" - вывод текста в двойных кавычках.
10. Охаракатеризуйте вывод информации на экран после выполнения команды ls с опцией l.  
    Будет выведена подробна информация о каждом файле:
    * тип файла
    * право доступа
    * число ссылок
    * владелец
    * размер
    * дата последней ревизии
    * имя каталога
11. Что такое относительный путь к файлу? Приведите примеры использования относительного и абсолютного пути при выполнении какой-либо команды.  
    Оносительный путь всегда задается относительно рабочего каталога или активного приложения.
    * cd ../../tmp/ - такая команда, вызванная из домашнего каталога переместит нас в директорию tmp/ (пример использования относительного пути).
    * cd /var/spool/ - независимо от того в какой директории мы сейчас находимся, эта команда всегда переместит нас в /var/spool/ (пример использования абсолютного пути).
12. Как получить информацию об интересующей вас команде?  
    Для этого существует команда man <команда>. Eсли мануалов для текущей команды не существует, можно попробовать получить краткую справку командой help. У многих команд есть флаг --help.
13. Какая клавиша или комбинация клавиш служит для автоматического дополнения вводимых команд?  
    Для автоматического дополнения команд можно использовать клавишу TAB.

# 6 Выводы

В этой работе мы разоборали и использовали на практике основные команды терминала Linux. Узнали некоторые нюансы их применения.

# Список литературы

1. Кулябов. Операционные системы. Москва: РУДН, 2016. 118 с.

2. Bash/Prompt customization [Электронный ресурс]. 2023. URL: <https://wiki.archlinux.org/title/Bash/Prompt_customization>.

3. How to Create Symbolic Link (Symlink) in Linux [Электронный ресурс]. 2023. URL: <https://phoenixnap.com/kb/symbolic-link-linux>.

4. Command-line interface [Электронный ресурс]. 2024. URL: <https://en.wikipedia.org/wiki/Command-line_interface>.

5. Escape character [Электронный ресурс]. 2024. URL: <https://en.wikipedia.org/wiki/Escape_character>.