

# **Отчёт по лабораторной работе №4**

**Основы интерфейса взаимодействия пользователя с системой Unix на  
уровне командной строки**

Тагиев Павел Фаикович

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>8</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>9</b>
4.1	Некоторые необходимые сведения . . . . .	9
4.2	Полное имя домашней директории . . . . .	10
4.3	Директория /tmp и ее содержимое . . . . .	11
4.4	Файл cron . . . . .	12
4.5	Домашний каталог . . . . .	13
4.6	Использование mkdir, rmdir и rm . . . . .	14
4.7	Рекурсивный вывод команды ls . . . . .	16
4.8	Сортировка вывода ls по времени последнего изменения . . . . .	17
4.9	Документация к команде cd . . . . .	19
4.10	Документация к команде pwd . . . . .	21
4.11	Документация к команде mkdir . . . . .	22
4.12	Документация к команде rmdir . . . . .	23
4.13	Документация к команде rm . . . . .	24
4.14	Модификация команд из history . . . . .	25
<b>5</b>	<b>Ответы на контрольные вопросы</b>	<b>28</b>
<b>6</b>	<b>Выводы</b>	<b>31</b>
	<b>Список литературы</b>	<b>32</b>

## Список иллюстраций

4.1	Мой prompt . . . . .	10
4.2	Полное имя домашней директории . . . . .	11
4.3	Директория tmp . . . . .	12
4.4	Каталог /var/spool/ . . . . .	13
4.5	Домашний каталог . . . . .	14
4.6	Создание и удаление каталогов . . . . .	15
4.7	Рекурсивный вывод . . . . .	16
4.8	Документация ls . . . . .	17
4.9	Документация ls, sort by time . . . . .	18
4.10	Сортировка вывода ls . . . . .	18
4.11	Команда help для cd . . . . .	19
4.12	Документация pwd . . . . .	21
4.13	Документация mkdir . . . . .	22
4.14	Документация rmdir . . . . .	23
4.15	Документация rm . . . . .	24
4.16	Использование history . . . . .	25
4.17	Использование модифицированных команд . . . . .	26

## Список таблиц

4.1	Символы определяющие тип файла . . . . .	12
4.2	Различия ссылок . . . . .	20

# 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 владельцем файлов и подкаталогов.
3. Выполнить 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 новых 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 указанного 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 информацию 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 построчно-го 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 используем команды:

`cd`, `pwd`, `ls`, `mkdir`, `rmdir`, `rm` и `man`.



## 4 Выполнение лабораторной работы

### 4.1 Некоторые необходимые сведения

Перед основной частью отчета, нужно уточнить некоторые моменты. Эту работу я выполнял на WSL с установленной ОС Ubuntu. В терминале я использую немного измененный prompt(рис. 4.1). 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 клавиш `Ctrl + l`. Так мне будет удобнее ссылаться на части терминала.

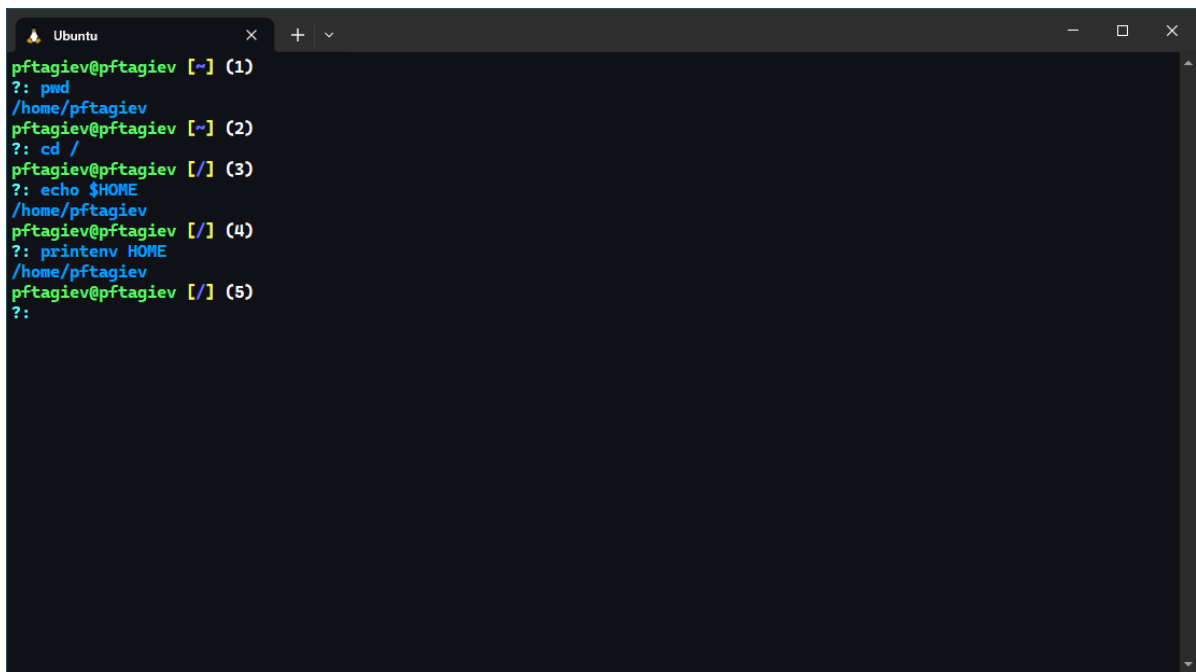
```
pftagiev@pftagiev [~] (1)
?: cd work/study/2023-2024/Операционные системы/os-intro/
pftagiev@pftagiev [~/work/study/2023-2024/Операционные системы/os-intro] [master f079d01] (2)
?: \
&: \
&: ls -F
CHANGELOG.md LICENSE README.en.md README.md labs/ presentation/ template/
COURSE Makefile README.git-flow.md config/ prepare project-personal/
pftagiev@pftagiev [~/work/study/2023-2024/Операционные системы/os-intro] [master f079d01] (3)
?: _
```

Рис. 4.1: 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. 4.1 его нужно поместить в файл `~/.bashrc`. Теперь мы можем перейти к основной части отчета.

## 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. 4.2.

A terminal window titled 'Ubuntu' with a dark background. It shows a series of commands and their outputs, numbered (1) through (5). The commands are: 'pwd', 'cd /', 'echo \$HOME', 'printenv HOME', and another 'pwd'. The outputs are: '/home/pftagiev', '/', '/home/pftagiev', '/home/pftagiev', and '/home/pftagiev' respectively. The prompt is 'pftagiev@pftagiev' and the cursor is at the end of the last command.

```
pftagiev@pftagiev [~] (1)
?: pwd
/home/pftagiev
pftagiev@pftagiev [~] (2)
?: cd /
pftagiev@pftagiev [/] (3)
?: echo $HOME
/home/pftagiev
pftagiev@pftagiev [/] (4)
?: printenv HOME
/home/pftagiev
pftagiev@pftagiev [/] (5)
?:
```

Рис. 4.2: 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. 4.3. Используем команду `ls`, мы видим 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.

```
Ubuntu
pftagiev@pftagiev [~] (1)
?: cd /tmp
pftagiev@pftagiev [/tmp] (2)
?: ls
snap-private-tmp
systemd-private-a9cd4a3ea7d046eeaacd0cd90c99c97b-apache2.service-Vm00J8
systemd-private-a9cd4a3ea7d046eeaacd0cd90c99c97b-systemd-logind.service-SiUNU2
systemd-private-a9cd4a3ea7d046eeaacd0cd90c99c97b-systemd-resolved.service-x57e3f
pftagiev@pftagiev [/tmp] (3)
?: ls -l
total 16
drwx----- 2 root root 4096 Apr  5 10:05 snap-private-tmp
drwx----- 3 root root 4096 Apr  5 10:05 systemd-private-a9cd4a3ea7d046eeaacd0cd90c99c97b-apache2.service-Vm00J8
drwx----- 3 root root 4096 Apr  5 10:05 systemd-private-a9cd4a3ea7d046eeaacd0cd90c99c97b-systemd-logind.service-SiUNU2
drwx----- 3 root root 4096 Apr  5 10:05 systemd-private-a9cd4a3ea7d046eeaacd0cd90c99c97b-systemd-resolved.service-x57e3f
pftagiev@pftagiev [/tmp] (4)
?: ls -la
..          .XIM-unix      systemd-private-a9cd4a3ea7d046eeaacd0cd90c99c97b-apache2.service-Vm00J8
.ICE-unix   .font-unix     systemd-private-a9cd4a3ea7d046eeaacd0cd90c99c97b-systemd-logind.service-SiUNU2
.Test-unix  snap-private-tmp systemd-private-a9cd4a3ea7d046eeaacd0cd90c99c97b-systemd-resolved.service-x57e3f
pftagiev@pftagiev [/tmp] (5)
?: ls -F
snap-private-tmp/
systemd-private-a9cd4a3ea7d046eeaacd0cd90c99c97b-apache2.service-Vm00J8/
systemd-private-a9cd4a3ea7d046eeaacd0cd90c99c97b-systemd-logind.service-SiUNU2/
systemd-private-a9cd4a3ea7d046eeaacd0cd90c99c97b-systemd-resolved.service-x57e3f/
pftagiev@pftagiev [/tmp] (6)
?: =
```

Рис. 4.3: Директория tmp

Посмотрим есть ли в этой папке скрытые файлы, добавив опцию -a. Как видно на рис. 4.3 они есть. Теперь, наконец, узнаем тип этих файлов добавив флаг -F, в соответствии с табл. 4.1 это каталоги.

Таблица 4.1: 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. 4.4

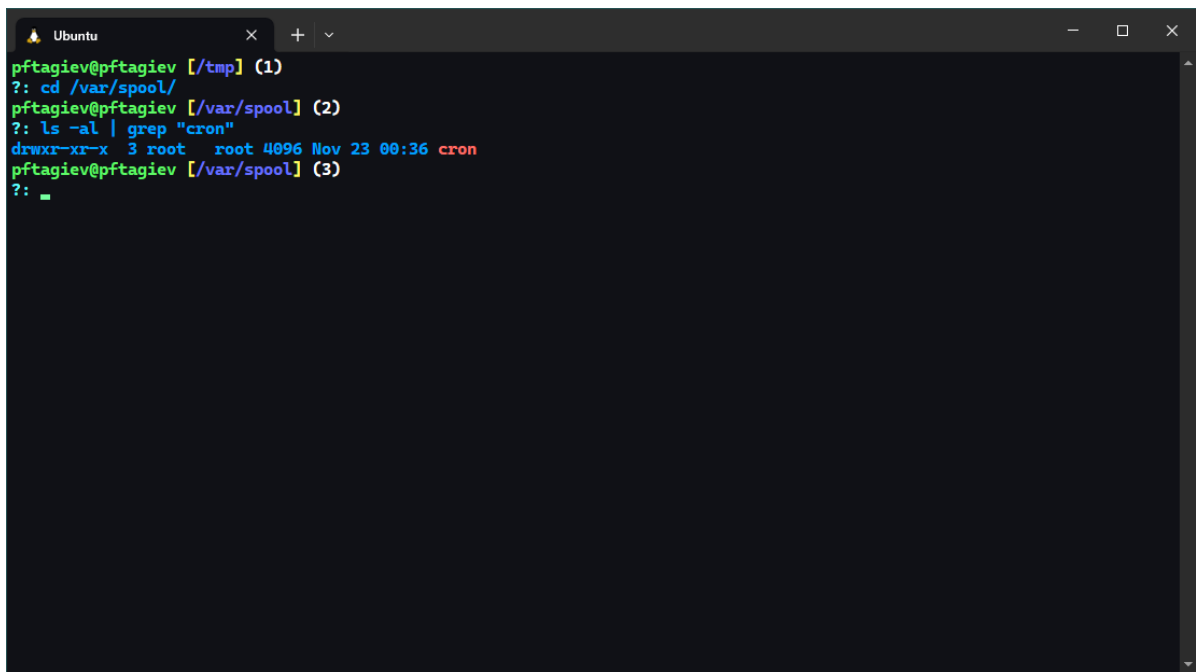
A terminal window titled 'Ubuntu' with a dark background. The user 'pftagiev' is at the prompt. The sequence of commands and outputs is as follows:  
1. Command: `cd /var/spool/` (1)  
Output: `?: cd /var/spool/`  
2. Command: `ls -al | grep "cron"` (2)  
Output: `drwxr-xr-x 3 root root 4096 Nov 23 00:36 cron`  
3. Command: `ls -al` (3)  
Output: `?:`  
The terminal window has standard Ubuntu window controls (minimize, maximize, close) in the top right corner.

Рис. 4.4: 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. 4.5 всеми файлами этой директории владеет пользователь `pftagiev`, а директорией выше уже пользователь `root`.

```
Ubuntu
pftagiev@pftagiev [/var/spool] (1)
?: cd
pftagiev@pftagiev [~] (2)
?: ls -alF
total 164
drwxr-xr-x 15 pftagiev pftagiev 4096 Apr  5 10:06 ./
drwxr-xr-x  3 root      root      4096 Mar 12 22:44 ../
-rw-r--r--  1 pftagiev pftagiev 3082 Apr  5 00:58 .bash_history
-rw-r--r--  1 pftagiev pftagiev 220  Mar 12 22:44 .bash_logout
-rw-r--r--  1 pftagiev pftagiev 4553 Apr  4 23:51 .bashrc
drwxr-xr-x  9 pftagiev pftagiev 4096 Mar 28 17:14 .cache/
drwxr-xr-x  6 pftagiev pftagiev 4096 Apr  4 18:29 .config/
drwxr-xr-x  3 pftagiev pftagiev 4096 Mar 13 10:21 .dotnet/
-rw-r--r--  1 pftagiev pftagiev 277  Mar 21 23:50 .gitconfig
drwxr-xr-x  4 pftagiev pftagiev 4096 Mar 27 10:55 .gnupg/
-rw-r--r--  1 pftagiev pftagiev   0  Mar 12 22:56 .hushlogin
-rw-r--r--  1 pftagiev pftagiev 195  Apr  4 23:02 .lesshtst
-rw-r--r--  1 pftagiev pftagiev   0  Mar 12 22:44 .motd_shown
-rw-r--r--  1 pftagiev pftagiev 844  Mar 15 22:56 .profile
-rw-r--r--  1 pftagiev pftagiev 120  Mar 20 00:16 .python_history
drwxr-xr-x  2 pftagiev pftagiev 4096 Mar 16 11:18 .ssh/
-rw-r--r--  1 pftagiev pftagiev   0  Mar 12 22:46 .sudo_as_admin_successful
drwxr-xr-x  3 pftagiev pftagiev 4096 Mar 18 12:17 .texlive2021/
-rw-r--r--  1 pftagiev pftagiev 9835 Apr  5 10:06 .viminfo
-rw-r--r--  1 pftagiev pftagiev 66   Apr  3 20:06 .vimrc
drwxr-xr-x  5 pftagiev pftagiev 4096 Mar 13 10:15 .vscode-server/
-rw-r--r--  1 pftagiev pftagiev 249  Apr  4 23:05 .wget-hsts
-rw-r--r--  1 pftagiev pftagiev 49076 Mar 15 18:17 .zcompdump
-rw-r--r--  1 pftagiev pftagiev 31   Mar 15 18:59 .zshrc
drwxr-xr-x  5 pftagiev pftagiev 4096 Apr  4 21:26 dir/
drwxr-xr-x  3 pftagiev pftagiev 4096 Apr  2 11:29 downloads/
drwxr-xr-x  4 pftagiev pftagiev 4096 Mar 13 10:22 go/
drwxr-xr-x  2 pftagiev pftagiev 4096 Apr  4 21:19 newdir/
drwxr-xr-x  7 pftagiev pftagiev 4096 Apr  2 11:40 repos/
drwxr-xr-x  3 pftagiev pftagiev 4096 Mar 15 20:04 work/
pftagiev@pftagiev [~] (3)
?: -
```

Рис. 4.5: Домашний каталог

## 4.6 Использование mkdir, rmdir и rm

Разберем что происходит на рис. 4.6, с помощью списка где каждый пункт соответствует промпт на изображении.

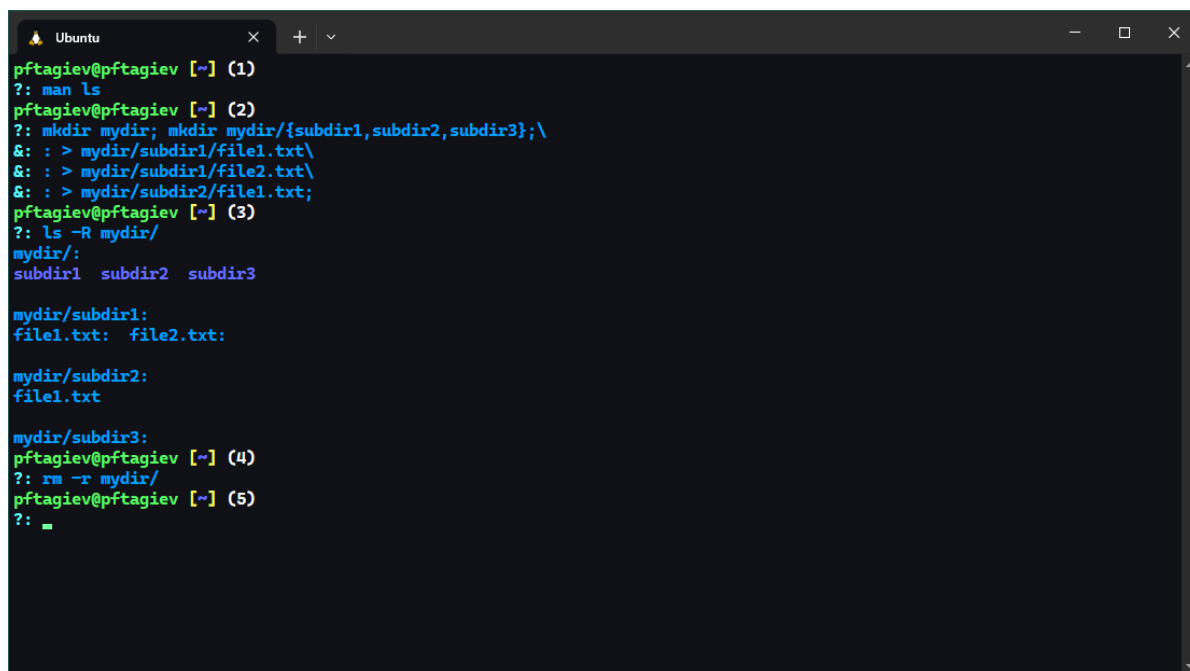
```
Ubuntu
pftagiev@pftagiev [~] (1)
?: mkdir -p newdir/morefun
pftagiev@pftagiev [~] (2)
?: ls newdir/
morefun
pftagiev@pftagiev [~] (3)
?: mkdir ./{letters,memos,misk}
pftagiev@pftagiev [~] (4)
?: ls
dir downloads go letters memos misk newdir repos work
pftagiev@pftagiev [~] (5)
?: rm -r letters memos misk
pftagiev@pftagiev [~] (6)
?: ls
dir downloads go newdir repos work
pftagiev@pftagiev [~] (7)
?: rm newdir/
rm: cannot remove 'newdir/': Is a directory
pftagiev@pftagiev [~] (8)
?: rm -r newdir/morefun/
pftagiev@pftagiev [~] (9)
?: ls newdir/
pftagiev@pftagiev [~] (10)
?: rmdir newdir/
pftagiev@pftagiev [~] (11)
?: ls
dir downloads go repos work
pftagiev@pftagiev [~] (12)
?:
```

Рис. 4.6: 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 конечный каталог был создан.
3. Создадим одной командой каталоги `letters`, `memos`, `misk`.
4. Проверим 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. 4.7.



```
pftagiev@pftagiev [~] (1)
?: man ls
pftagiev@pftagiev [~] (2)
?: mkdir mydir; mkdir mydir/{subdir1,subdir2,subdir3};\
&: : > mydir/subdir1/file1.txt\
&: : > mydir/subdir1/file2.txt\
&: : > mydir/subdir2/file1.txt;
pftagiev@pftagiev [~] (3)
?: ls -R mydir/
mydir/:
subdir1  subdir2  subdir3

mydir/subdir1:
file1.txt  file2.txt:

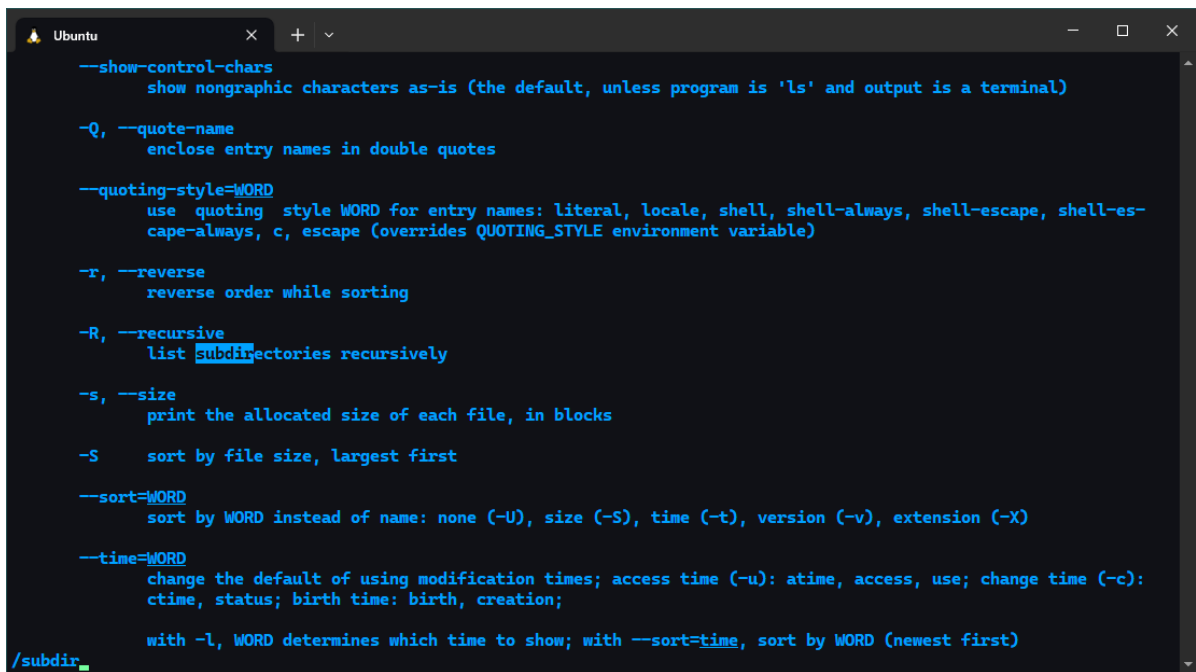
mydir/subdir2:
file1.txt

mydir/subdir3:
pftagiev@pftagiev [~] (4)
?: rm -r mydir/
pftagiev@pftagiev [~] (5)
?: _
```

Рис. 4.7: Рекурсивный вывод

Откроется документация к команде `ls`. Воспользуемся поиском по документации введя `/то_что_ищем`. Как видно на рис. 4.8 нам нужна опция `-R`. На рис. 4.7 в (3) можно увидеть использование этой опции.





```
--show-control-chars
  show nongraphic characters as-is (the default, unless program is 'ls' and output is a terminal)

-Q, --quote-name
  enclose entry names in double quotes

--quoting-style=WORD
  use quoting style WORD for entry names: literal, locale, shell, shell-always, shell-es-
  cape-always, c, escape (overrides QUOTING_STYLE environment variable)

-r, --reverse
  reverse order while sorting

-R, --recursive
  list subdirectories recursively

-s, --size
  print the allocated size of each file, in blocks

-S
  sort by file size, largest first

--sort=WORD
  sort by WORD instead of name: none (-U), size (-S), time (-t), version (-v), extension (-X)

--time=WORD
  change the default of using modification times; access time (-u): atime, access, use; change time (-c):
  ctime, status; birth time: birth, creation;

  with -l, WORD determines which time to show; with --sort=time, sort by WORD (newest first)

/subdir_
```

Рис. 4.8: 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. 4.9). Исходя из документации, чтобы отсортировать вывод `ls` по времени последнего изменения нужно использовать ключ `-t`, пример использования этого ключа можно увидеть на рис. 4.10.

```
Ubuntu
-S    sort by file size, largest first

--sort=WORD
      sort by WORD instead of name: none (-U), size (-S), time (-t), version (-v), extension (-X)

--time=WORD
      change the default of using modification times; access time (-u): atime, access, use; change time (-c):
      ctime, status; birth time: birth, creation;

      with -l, WORD determines which time to show; with --sort=time, sort by WORD (newest first)

--time-style=TIME_STYLE
      time/date format with -l; see TIME_STYLE below

-t    sort by time, newest first; see --time

-T, --tabsize=COLS
      assume tab stops at each COLS instead of 8

-u    with -lt: sort by, and show, access time; with -l: show access time and sort by name; otherwise: sort
      by access time, newest first

-U    do not sort; list entries in directory order

-v    natural sort of (version) numbers within text

-w, --width=COLS
      set output width to COLS. 0 means no limit

/sort by time
```

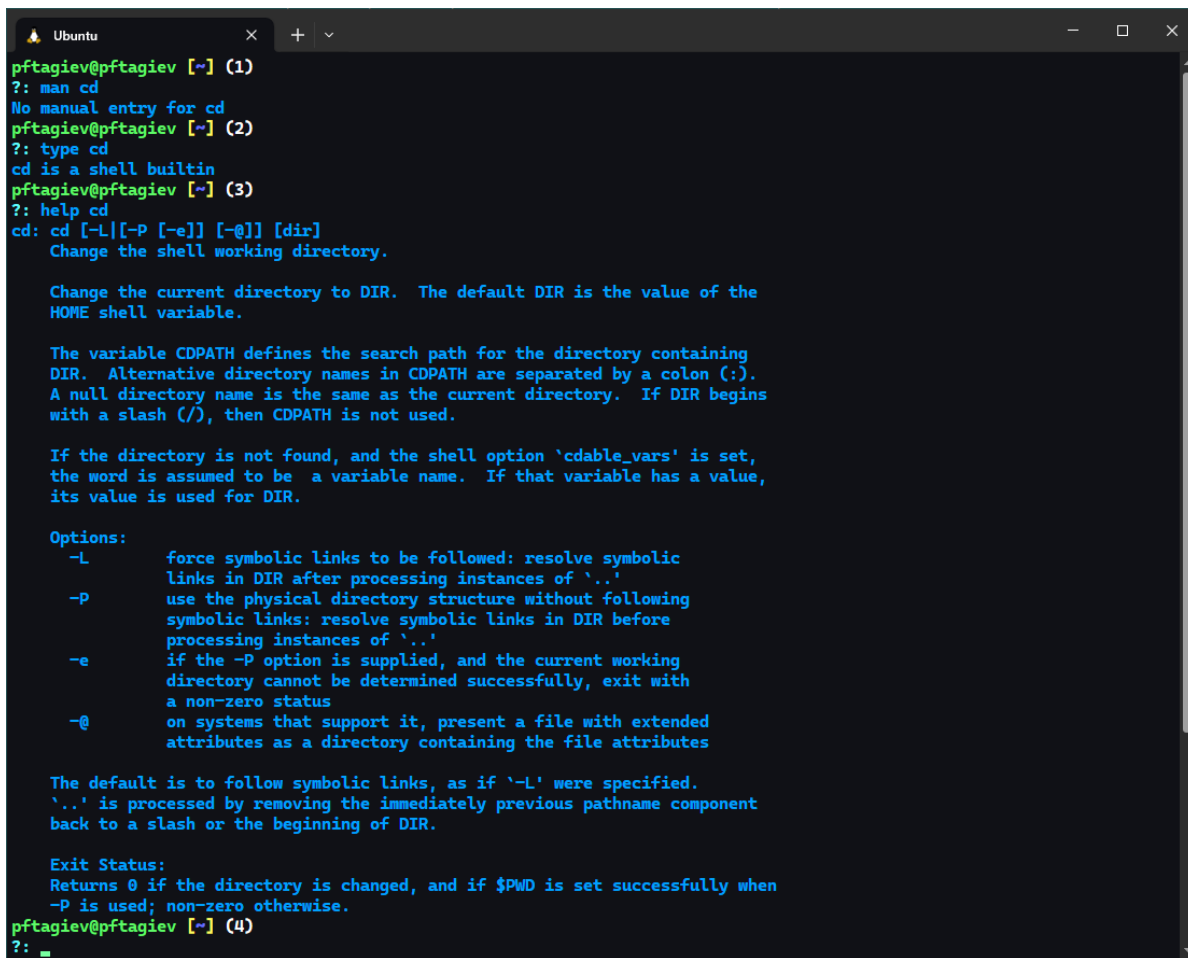
Рис. 4.9: Документация ls, sort by time

```
Ubuntu
pftagiev@pftagiev [~] (1)
?: man ls
pftagiev@pftagiev [~] (2)
?: ls -alt
total 160
drwxr-xr-x 14 pftagiev pftagiev 4096 Apr  5 10:38 .
-rw-r--r--  1 pftagiev pftagiev  225 Apr  5 10:38 .lessht
-rw-r--r--  1 pftagiev pftagiev 3319 Apr  5 10:26 .bash_history
-rw-r--r--  1 pftagiev pftagiev 9835 Apr  5 10:06 .viminfo
-rw-r--r--  1 pftagiev pftagiev 4553 Apr  4 23:51 .bashrc
-rw-r--r--  1 pftagiev pftagiev  249 Apr  4 23:05 .wget-hsts
drwxr-xr-x  5 pftagiev pftagiev 4096 Apr  4 21:26 dir
drwxr-xr-x  6 pftagiev pftagiev 4096 Apr  4 18:29 .config
-rw-r--r--  1 pftagiev pftagiev   66 Apr  3 20:06 .vimrc
drwxr-xr-x  7 pftagiev pftagiev 4096 Apr  2 11:40 repos
drwxr-xr-x  3 pftagiev pftagiev 4096 Apr  2 11:29 downloads
drwxr-xr-x  9 pftagiev pftagiev 4096 Mar 28 17:14 .cache
drwxr-xr-x  4 pftagiev pftagiev 4096 Mar 27 10:55 .gnupg
-rw-r--r--  1 pftagiev pftagiev  277 Mar 21 23:50 .gitconfig
-rw-r--r--  1 pftagiev pftagiev  120 Mar 20 00:16 .python_history
drwxr-xr-x  3 pftagiev pftagiev 4096 Mar 18 12:17 .texlive2021
drwxr-xr-x  2 pftagiev pftagiev 4096 Mar 16 11:18 .ssh
-rw-r--r--  1 pftagiev pftagiev  844 Mar 15 22:56 .profile
drwxr-xr-x  3 pftagiev pftagiev 4096 Mar 15 20:04 work
-rw-r--r--  1 pftagiev pftagiev   31 Mar 15 18:59 .zshrc
-rw-r--r--  1 pftagiev pftagiev 49076 Mar 15 18:17 .zcompdump
drwxr-xr-x  4 pftagiev pftagiev 4096 Mar 13 10:22 go
drwxr-xr-x  3 pftagiev pftagiev 4096 Mar 13 10:21 .dotnet
drwxr-xr-x  5 pftagiev pftagiev 4096 Mar 13 10:15 .vscode-server
-rw-r--r--  1 pftagiev pftagiev    0 Mar 12 22:56 .hushlogin
-rw-r--r--  1 pftagiev pftagiev    0 Mar 12 22:46 .sudo_as_admin_successful
-rw-r--r--  1 pftagiev pftagiev    0 Mar 12 22:44 .motd_shown
-rw-r--r--  1 pftagiev pftagiev  220 Mar 12 22:44 .bash_logout
drwxr-xr-x  3 root    root    4096 Mar 12 22:44 ..
pftagiev@pftagiev [~] (3)
?:
```

Рис. 4.10: Сортировка вывода ls

## 4.9 Документация к команде cd

К сожалению, так как команда `cd` является встроенной для нее нельзя получить документацию через утилиту `man` (рис. 4.11 prompt (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`.



```
pftagiev@pftagiev [~] (1)
?: man cd
No manual entry for cd
pftagiev@pftagiev [~] (2)
?: type cd
cd is a shell builtin
pftagiev@pftagiev [~] (3)
?: help cd
cd: cd [-L|[-P [-e]] [-@]] [dir]
    Change the shell working directory.

    Change the current directory to DIR.  The default DIR is the value of the
    HOME shell variable.

    The variable CDPATH defines the search path for the directory containing
    DIR.  Alternative directory names in CDPATH are separated by a colon (:).
    A null directory name is the same as the current directory.  If DIR begins
    with a slash (/), then CDPATH is not used.

    If the directory is not found, and the shell option 'cdable_vars' is set,
    the word is assumed to be a variable name.  If that variable has a value,
    its value is used for DIR.

Options:
  -L      force symbolic links to be followed: resolve symbolic
          links in DIR after processing instances of '..'
  -P      use the physical directory structure without following
          symbolic links: resolve symbolic links in DIR before
          processing instances of '..'
  -e      if the -P option is supplied, and the current working
          directory cannot be determined successfully, exit with
          a non-zero status
  -@      on systems that support it, present a file with extended
          attributes as a directory containing the file attributes

The default is to follow symbolic links, as if '-L' were specified.
'..' is processed by removing the immediately previous pathname component
back to a slash or the beginning of DIR.

Exit Status:
Returns 0 if the directory is changed, and if $PWD is set successfully when
-P is used; non-zero otherwise.
pftagiev@pftagiev [~] (4)
?:
```

Рис. 4.11: 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. 4.2 можно увидеть их различия.

Таблица 4.2: Различия ссылок

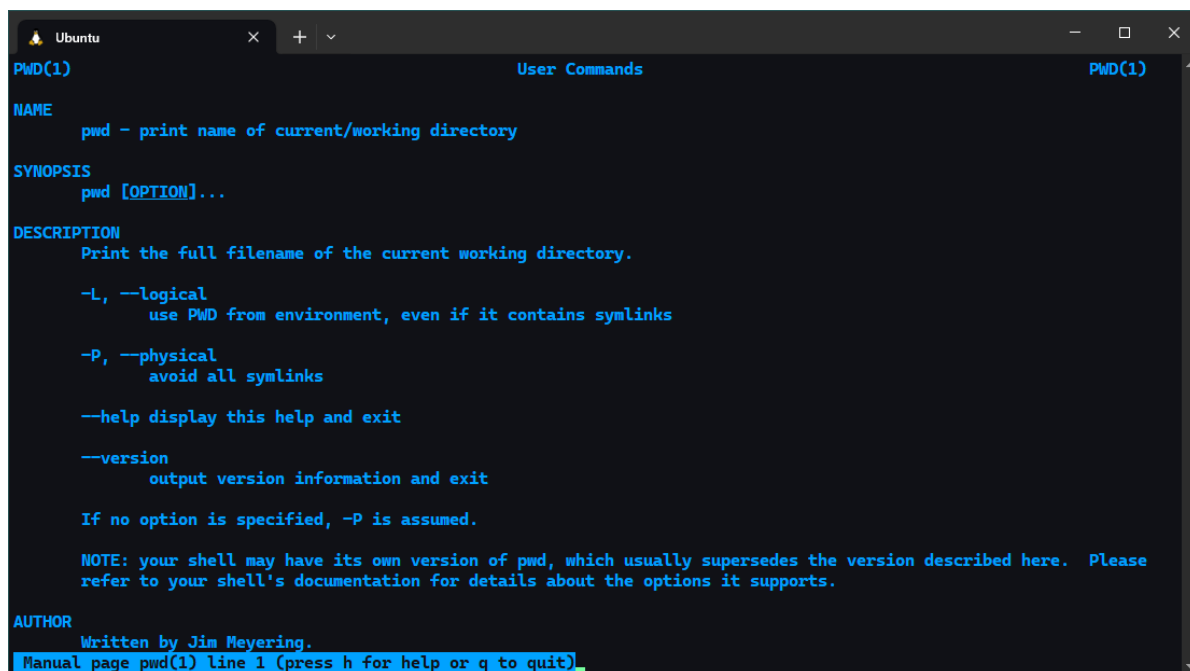
SoftLink	HardLink
Ссылается как на файл так и на каталог	Ссылается только на файл
Просто следуют по пути до файла или каталога	Указывает на фактические данные файла в томе хранения
Указывает на файлы и каталоги как на локальных так и на внешних томах	Может указывать на файлы только в локальном томе
Не 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. 4.11).

- `-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



```
PwD(1) User Commands PwD(1)
NAME
  pwd - print name of current/working directory
SYNOPSIS
  pwd [OPTION]...
DESCRIPTION
  Print the full filename of the current working directory.

  -L, --logical
        use PwD from environment, even if it contains symlinks

  -P, --physical
        avoid all symlinks

  --help display this help and exit

  --version
        output version information and exit

  If no option is specified, -P is assumed.

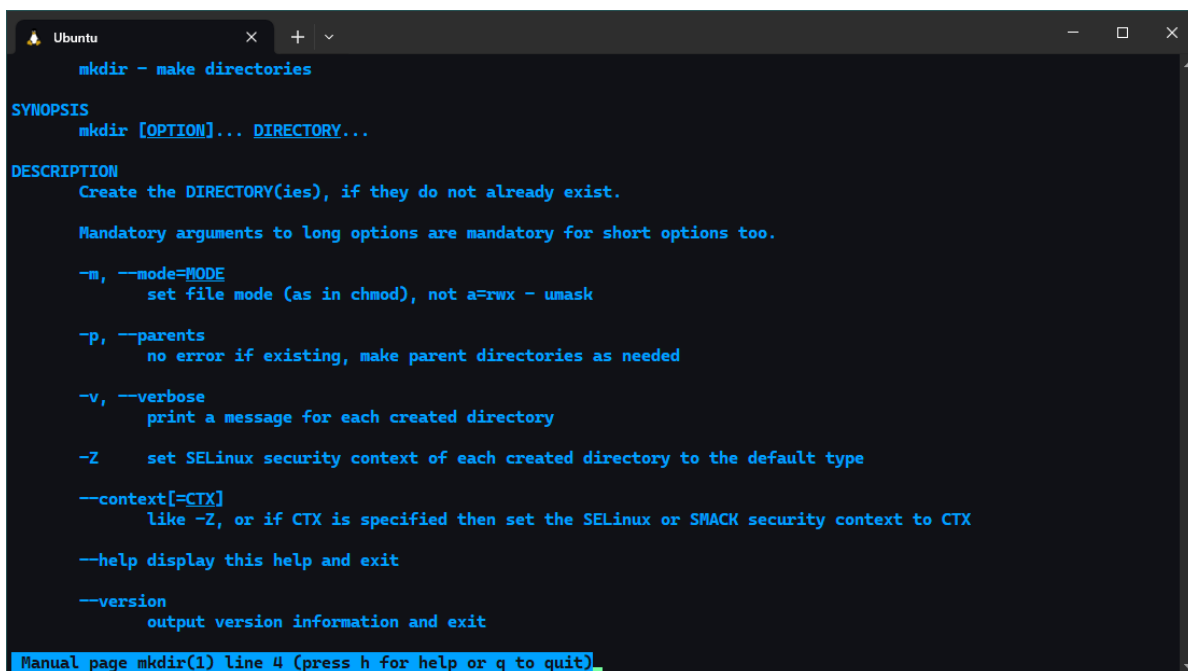
  NOTE: your shell may have its own version of pwd, which usually supersedes the version described here. Please
  refer to your shell's documentation for details about the options it supports.
AUTHOR
  Written by Jim Meyering.
Manual page pwd(1) line 1 (press h for help or q to quit)
```

Рис. 4.12: Документация pwd

Команда `pwd` выводит в терминал текущую рабочую директорию. Разберем ее основные флаги, которые показаны на рис. 4.12.

- `-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



```
Ubuntu
mkdir - make directories

SYNOPSIS
mkdir [OPTION]... DIRECTORY...

DESCRIPTION
Create the DIRECTORY(ies), if they do not already exist.

Mandatory arguments to long options are mandatory for short options too.

-m, --mode=MODE
    set file mode (as in chmod), not a=rwx - umask

-p, --parents
    no error if existing, make parent directories as needed

-v, --verbose
    print a message for each created directory

-Z      set SELinux security context of each created directory to the default type

--context[=CTX]
    like -Z, or if CTX is specified then set the SELinux or SMACK security context to CTX

--help display this help and exit

--version
    output version information and exit

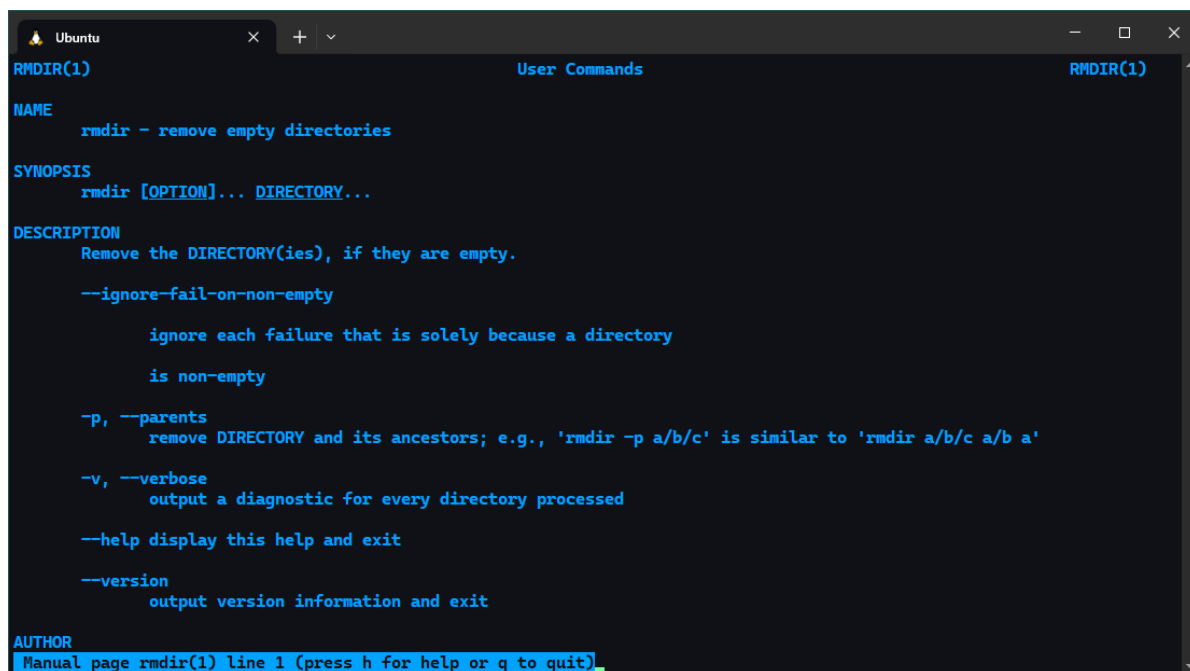
Manual page mkdir(1) line 4 (press h for help or q to quit)
```

Рис. 4.13: Документация mkdir

`mkdir` - создает каталог. Ее флаги можно увидеть на рис. 4.13, 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



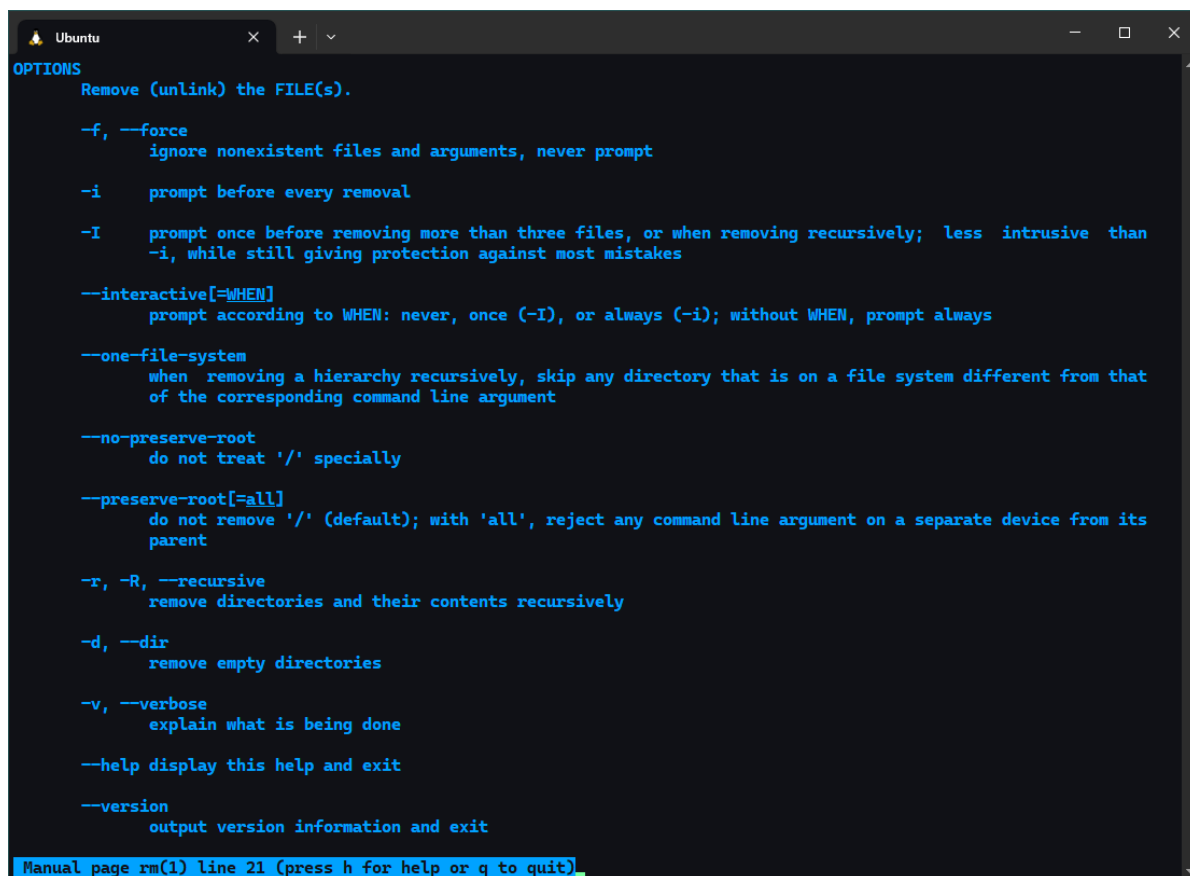
```
RMDIR(1) User Commands RMDIR(1)
NAME
  rmdir - remove empty directories
SYNOPSIS
  rmdir [OPTION]... DIRECTORY...
DESCRIPTION
  Remove the DIRECTORY(ies), if they are empty.
  --ignore-fail-on-non-empty
      ignore each failure that is solely because a directory
      is non-empty
  -p, --parents
      remove DIRECTORY and its ancestors; e.g., 'rmdir -p a/b/c' is similar to 'rmdir a/b/c a/b a'
  -v, --verbose
      output a diagnostic for every directory processed
  --help
      display this help and exit
  --version
      output version information and exit
AUTHOR
  Manual page rmdir(1) line 1 (press h for help or q to quit)
```

Рис. 4.14: Документация rmdir

`rmdir` - удаляет пустые каталоги (рис. 4.14).

- `--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

A screenshot of a terminal window showing the manual page for the 'rm' command. The window title is 'Ubuntu'. The content is the 'OPTIONS' section of the man page, listing various flags and their descriptions. The text is as follows:

```
OPTIONS
Remove (unlink) the FILE(s).

-f, --force
    ignore nonexistent files and arguments, never prompt

-i
    prompt before every removal

-I
    prompt once before removing more than three files, or when removing recursively; less intrusive than
    -i, while still giving protection against most mistakes

--interactive[=WHEN]
    prompt according to WHEN: never, once (-I), or always (-i); without WHEN, prompt always

--one-file-system
    when removing a hierarchy recursively, skip any directory that is on a file system different from that
    of the corresponding command line argument

--no-preserve-root
    do not treat '/' specially

--preserve-root[=all]
    do not remove '/' (default); with 'all', reject any command line argument on a separate device from its
    parent

-r, -R, --recursive
    remove directories and their contents recursively

-d, --dir
    remove empty directories

-v, --verbose
    explain what is being done

--help
    display this help and exit

--version
    output version information and exit

Manual page rm(1) line 21 (press h for help or q to quit)
```

Рис. 4.15: Документация rm

Команда `rm` - удаляет файлы и каталоги(если применить специальный флаг). Документация полученная командой `man` на рис. 4.15, ниже перевод.

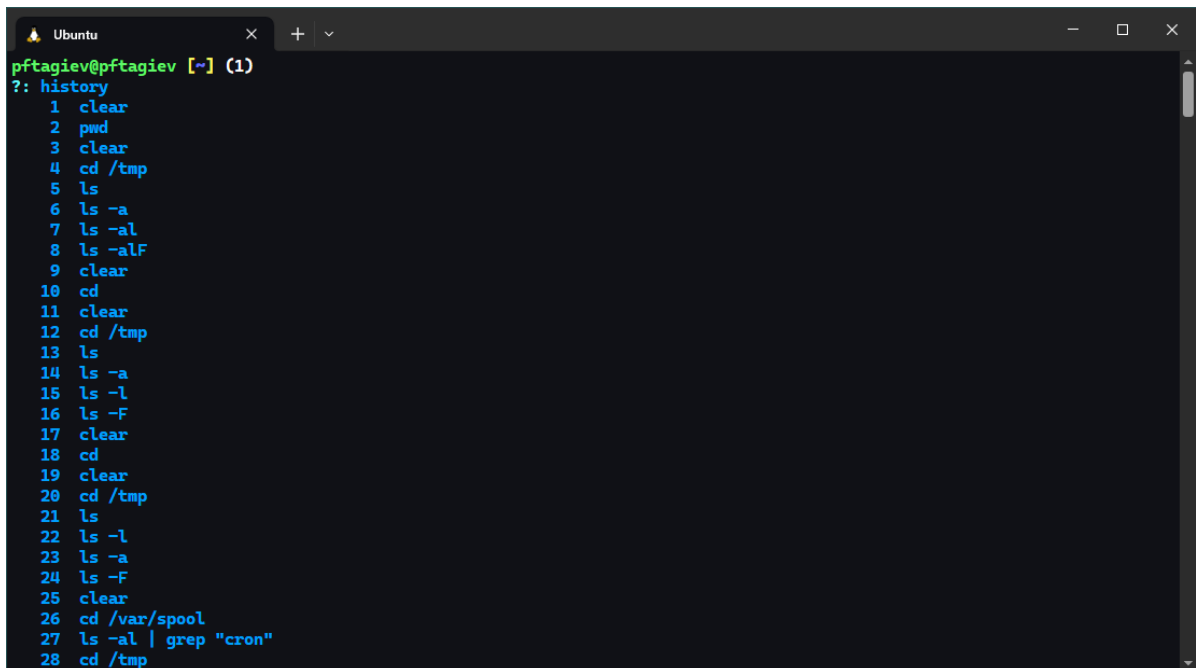
- `-f` - Игнорировать несуществующие файлы и аргументы. Никогда не выдавать запросы на подтверждение удаления.
- `-i` - Выводить запрос на подтверждение удаления каждого файла.
- `-I` - Выдать один запрос на подтверждение удаления всех файлов, если удаляется больше трех файлов или используется рекурсивное 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. 4.16.

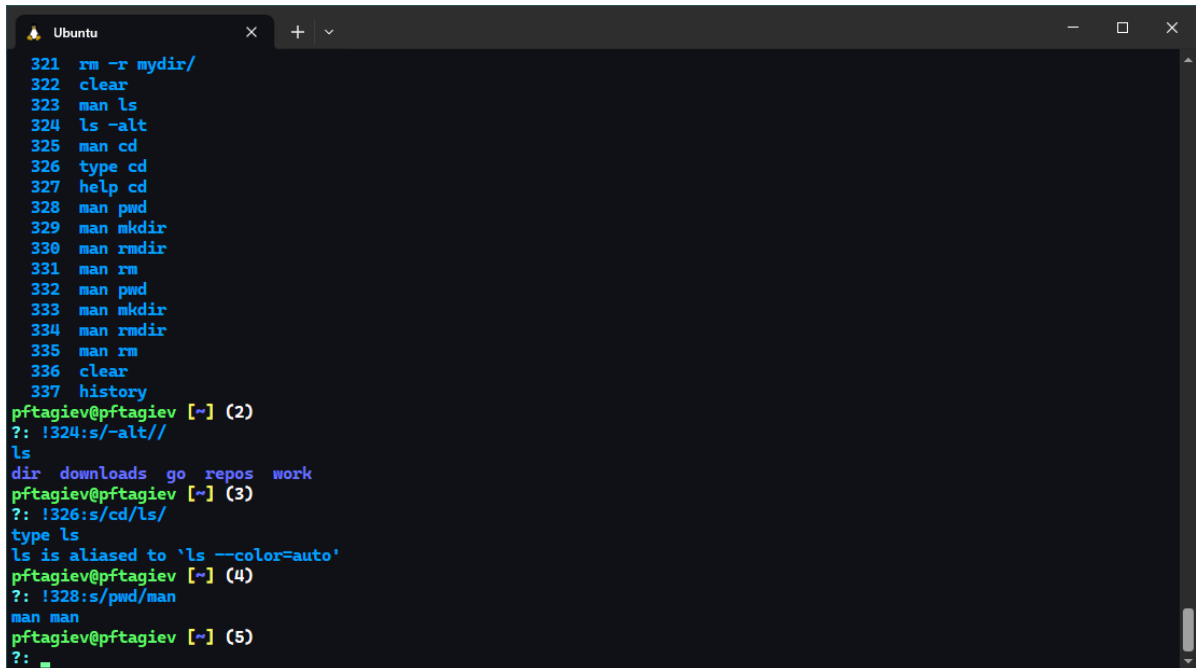


```

pftagiev@pftagiev [~] (1)
?: history
1 clear
2 pwd
3 clear
4 cd /tmp
5 ls
6 ls -a
7 ls -al
8 ls -alF
9 clear
10 cd
11 clear
12 cd /tmp
13 ls
14 ls -a
15 ls -l
16 ls -F
17 clear
18 cd
19 clear
20 cd /tmp
21 ls
22 ls -l
23 ls -a
24 ls -F
25 clear
26 cd /var/spool
27 ls -al | grep "cron"
28 cd /tmp
  
```

Рис. 4.16: Использование history

Вызовем модифицированные команды из буфера команд как показано на рис. 4.17 в prompt (2) - (4).



```
321 rm -r mydir/
322 clear
323 man ls
324 ls -alt
325 man cd
326 type cd
327 help cd
328 man pwd
329 man mkdir
330 man rmdir
331 man rm
332 man pwd
333 man mkdir
334 man rmdir
335 man rm
336 clear
337 history
pftagiev@pftagiev [~] (2)
?: !324:s/~alt//
ls
dir downloads go repos work
pftagiev@pftagiev [~] (3)
?: !326:s/cd/ls/
type ls
ls is aliased to `ls --color=auto'
pftagiev@pftagiev [~] (4)
?: !328:s/pwd/man
man man
pftagiev@pftagiev [~] (5)
?:
```

Рис. 4.17: Использование модифицированных команд

---

**Листинг 4.1** 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="\[\e[1;96m\]&: \[\e[0m\]"
```

---

## 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. 4.3 в prompt (5).

### 4. Каким образом отобразить информацию о скрытых файлах? Приведите примеры.

Для этого нужно использовать команду `ls` с флагом `-a`, пример уже встречался в отчете на рис. 4.3 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. 4.17.

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. Охарактеризуйте вывод информации на экран после выполнения команды `ls` с опцией `l`.

Будет выведена подробная 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.

Относительный 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>`. Если 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 разобрали 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](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](https://en.wikipedia.org/wiki/Command-line_interface).
5. Escape character [Электронный ресурс]. 2024. URL: [https://en.wikipedia.org/wiki/Escape\\_character](https://en.wikipedia.org/wiki/Escape_character).