

Отчёт по лабораторной работе №9

Командная оболочка Midnight Commander

Фатима Халилова

Содержание

1 Цель работы	5
2 Выполнение лабораторной работы	6
3 Вывод	41
4 Контрольные вопросы	42

Список иллюстраций

2.1 Запуск mc	7
2.2 Выделение	8
2.3 Отмена	9
2.4 Копирование	10
2.5 Перемещение	11
2.6 Информация	12
2.7 Быстрый просмотр	13
2.8 Информация	14
2.9 Дерево каталогов	15
2.10 Просмотр содержимого текстового файла	16
2.11 Отредактируем содержимое текстового файла без сохранения результатов	17
2.12 Создание каталога	18
2.13 Копирование в файлов в созданный каталог	19
2.14 Поиск файлов	20
2.15 История команд	21
2.16 Переход в домашний каталог	22
2.17 Просмотр файла расширений	23
2.18 Просмотр файла меню	24
2.19 Конфигурация	25
2.20 Внешний вид	26
2.21 Настройки панелей	27
2.22 Подтверждение	28
2.23 Оформление	29
2.24 Кодировка символов	30
2.25 Распознавание клавиш	31
2.26 Файл с текстом	32
2.27 Файл с текстом	33
2.28 Копирование фрагмента	34
2.29 Сохранение	35
2.30 Отмена	36
2.31 Переход в конец файла	37
2.32 Переход в начало файла	38
2.33 Файл с программой	39
2.34 Цветовыделение синтаксиса	40

Список таблиц

1 Цель работы

Освоение основных возможностей командной оболочки Midnight Commander.
Приобретение навыков практической работы по просмотру каталогов и файлов; манипуляций с ними.

2 Выполнение лабораторной работы

1 Изучим информацию о mc при помощи справки man. Воспользуемся справкой и узнаем что для того чтобы войти в командную оболочку мы должны ввести в командной строке mc.

2 Запустим из командной строки mc.

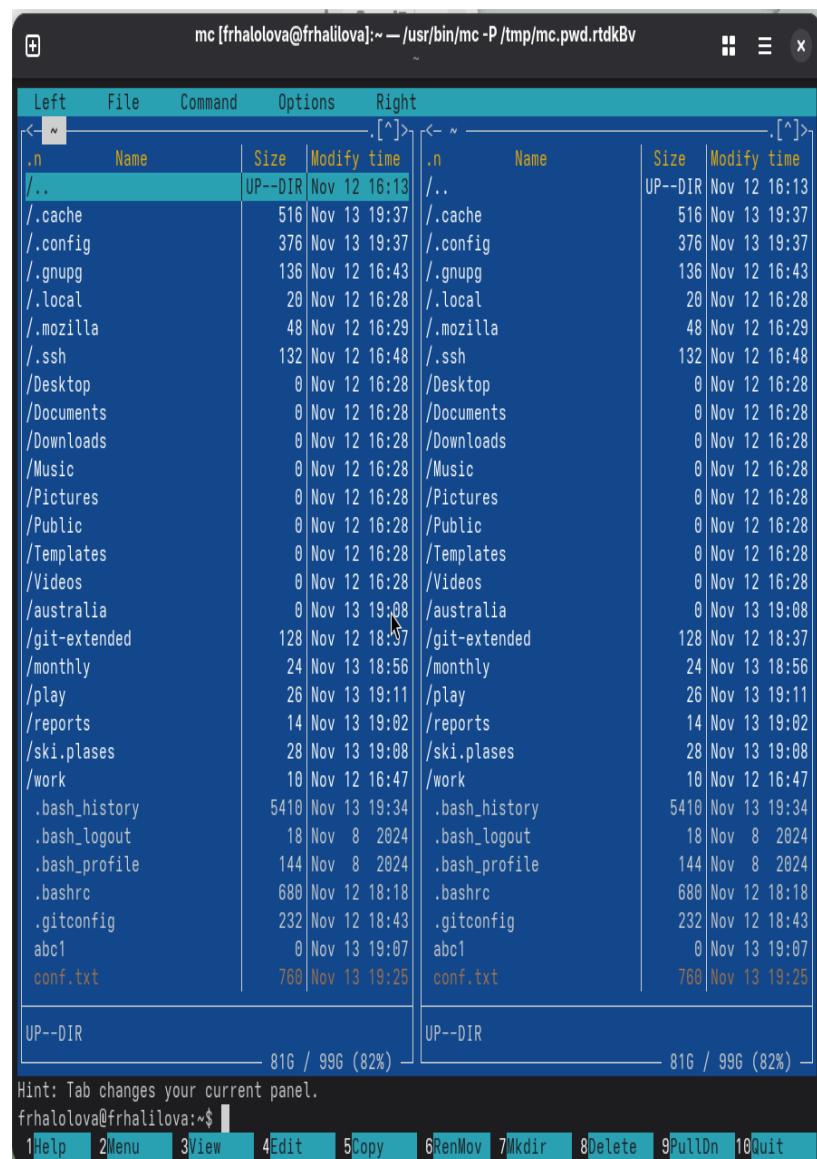


Рисунок 2.1: Запуск mc

3 Выполните несколько операций в mc, используя управляющие клавиши

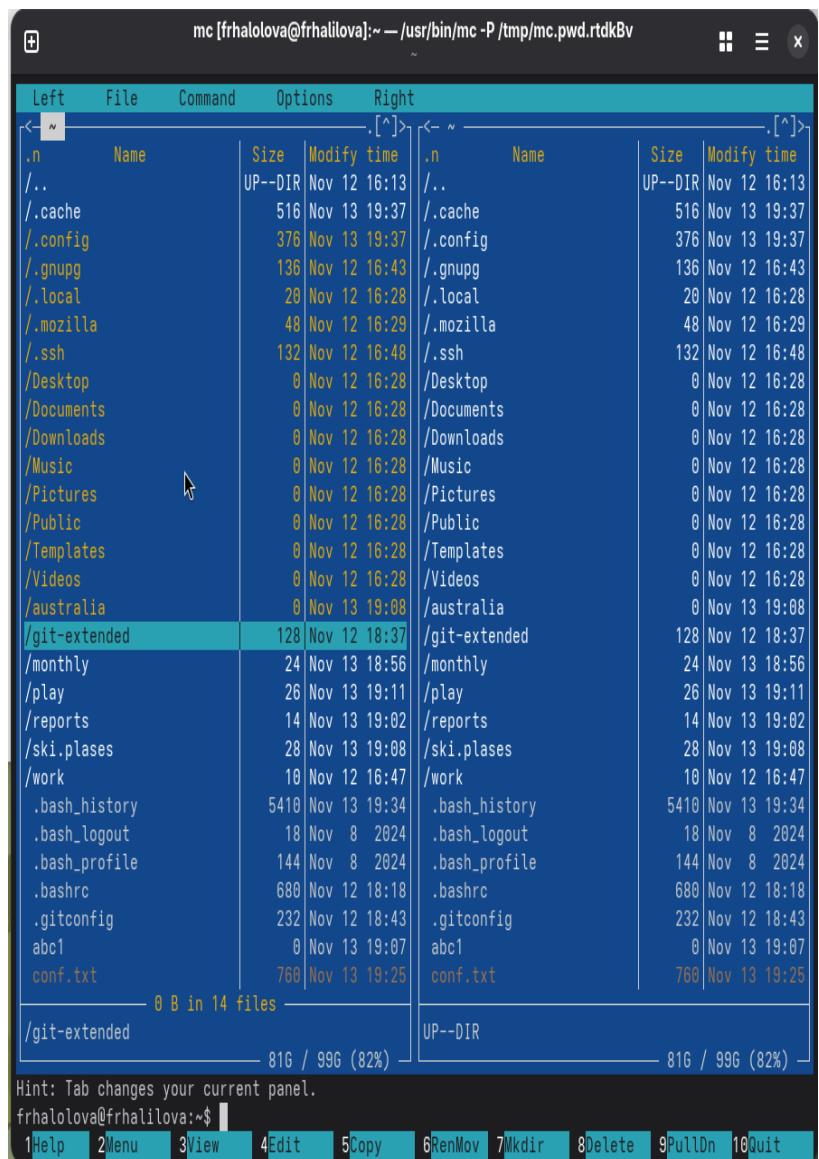


Рисунок 2.2: Выделение

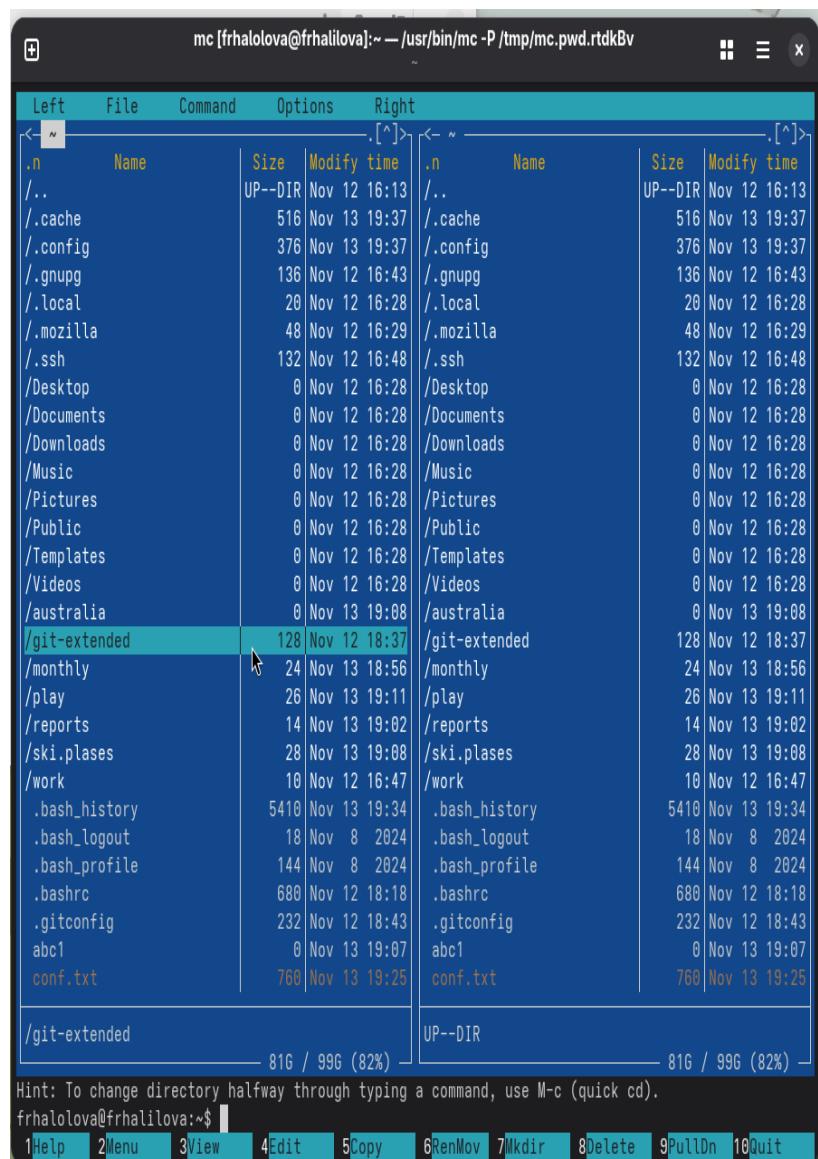


Рисунок 2.3: Отмена

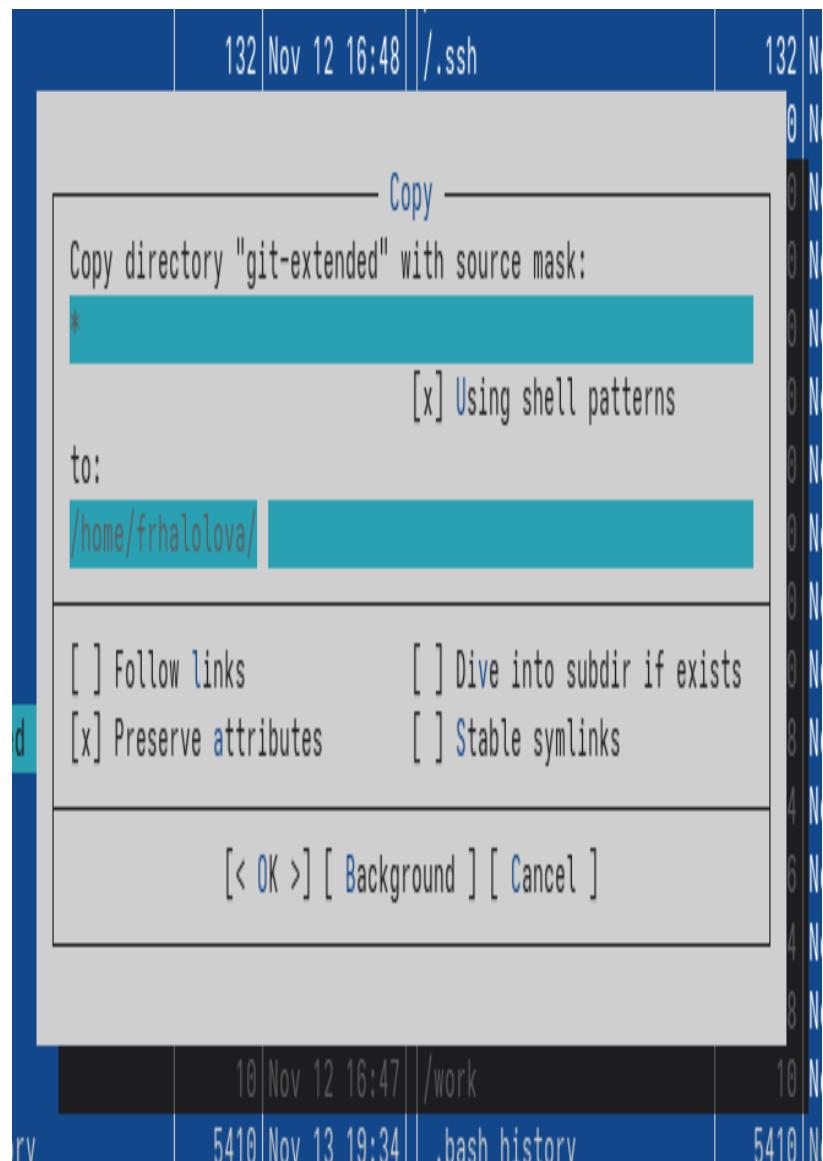


Рисунок 2.4: Копирование

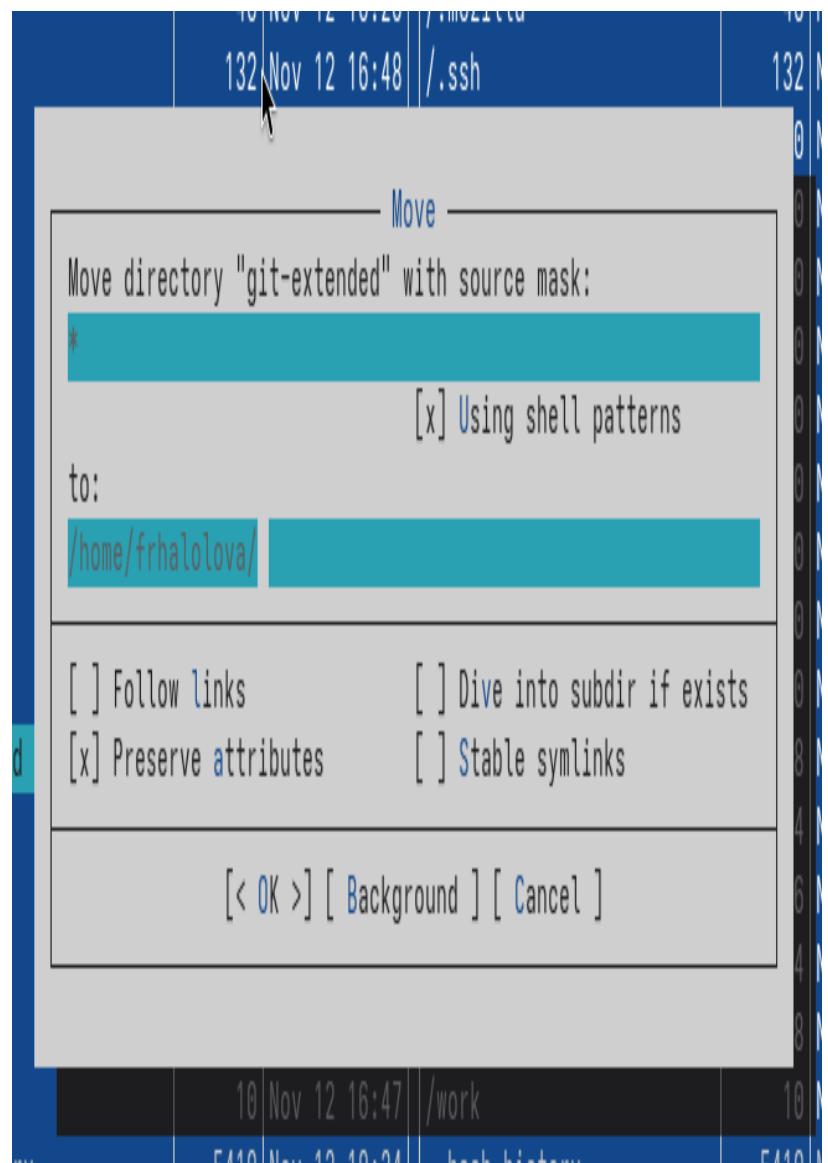


Рисунок 2.5: Перемещение

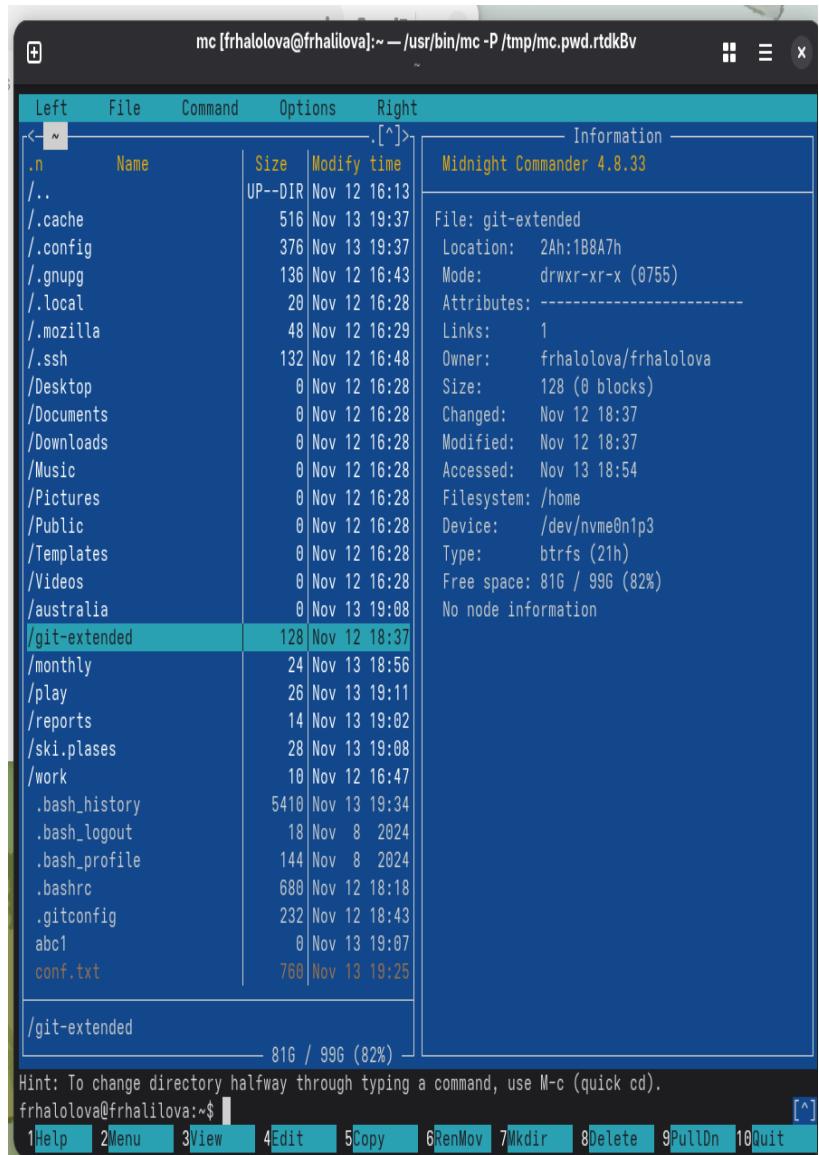


Рисунок 2.6: Информация

4 Выполните основные команды меню левой панели.

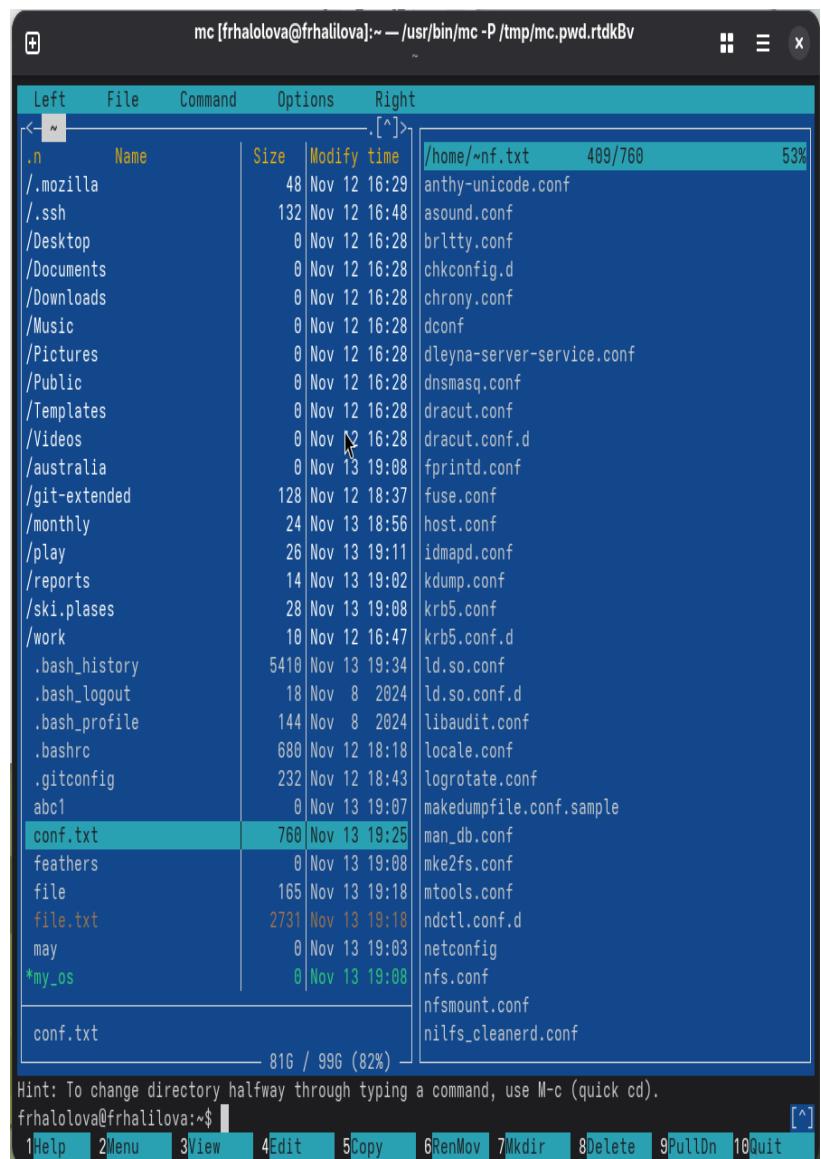


Рисунок 2.7: Быстрый просмотр

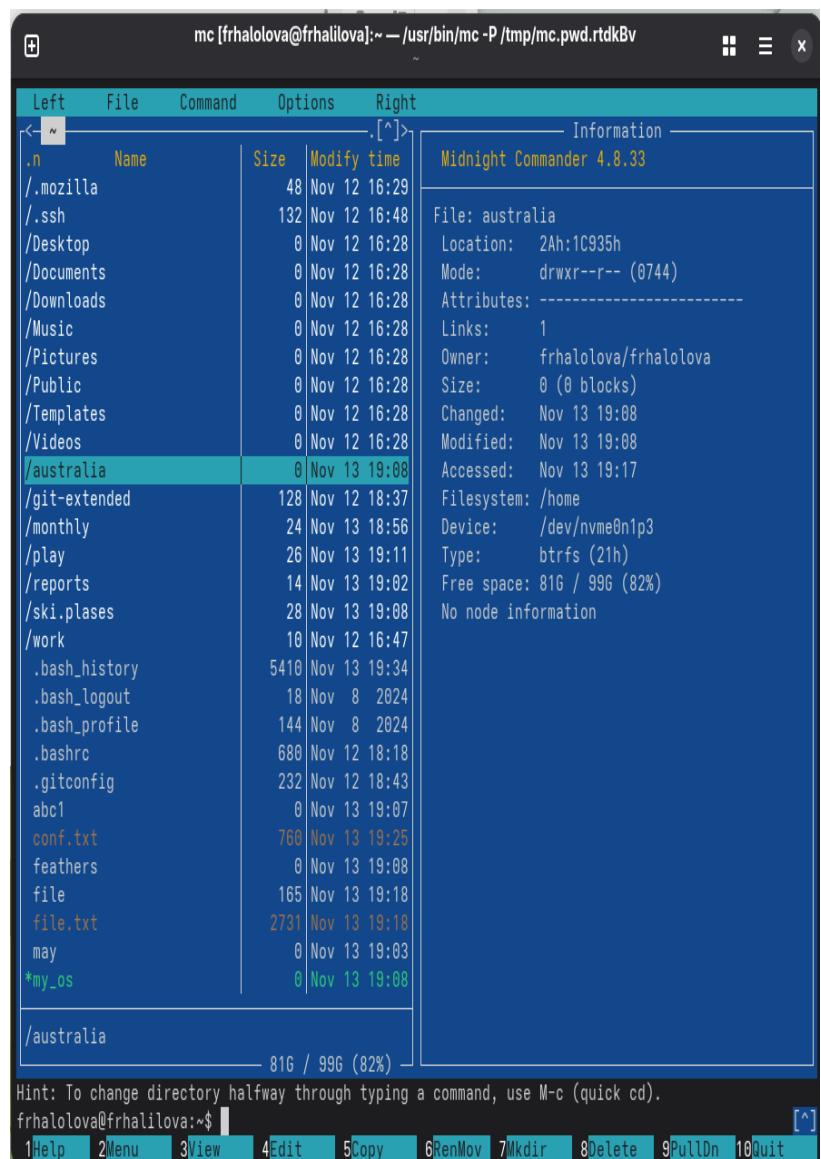


Рисунок 2.8: Информация

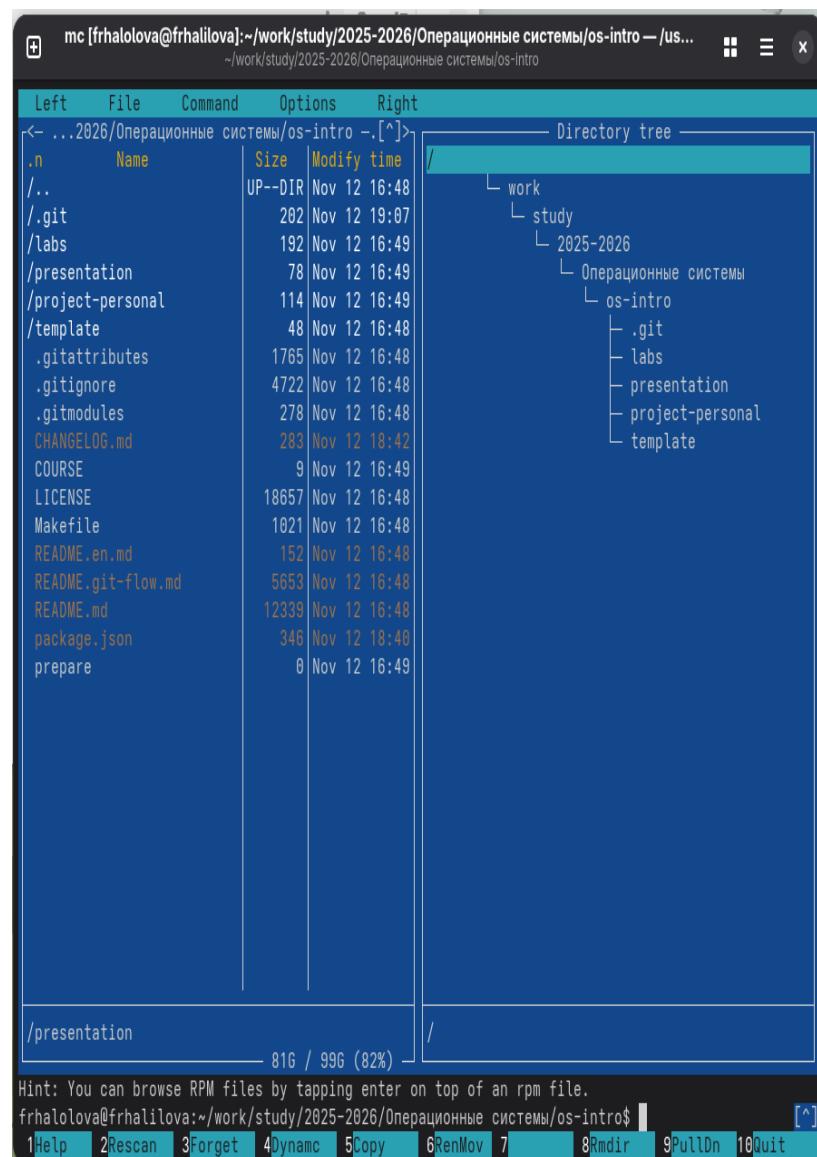
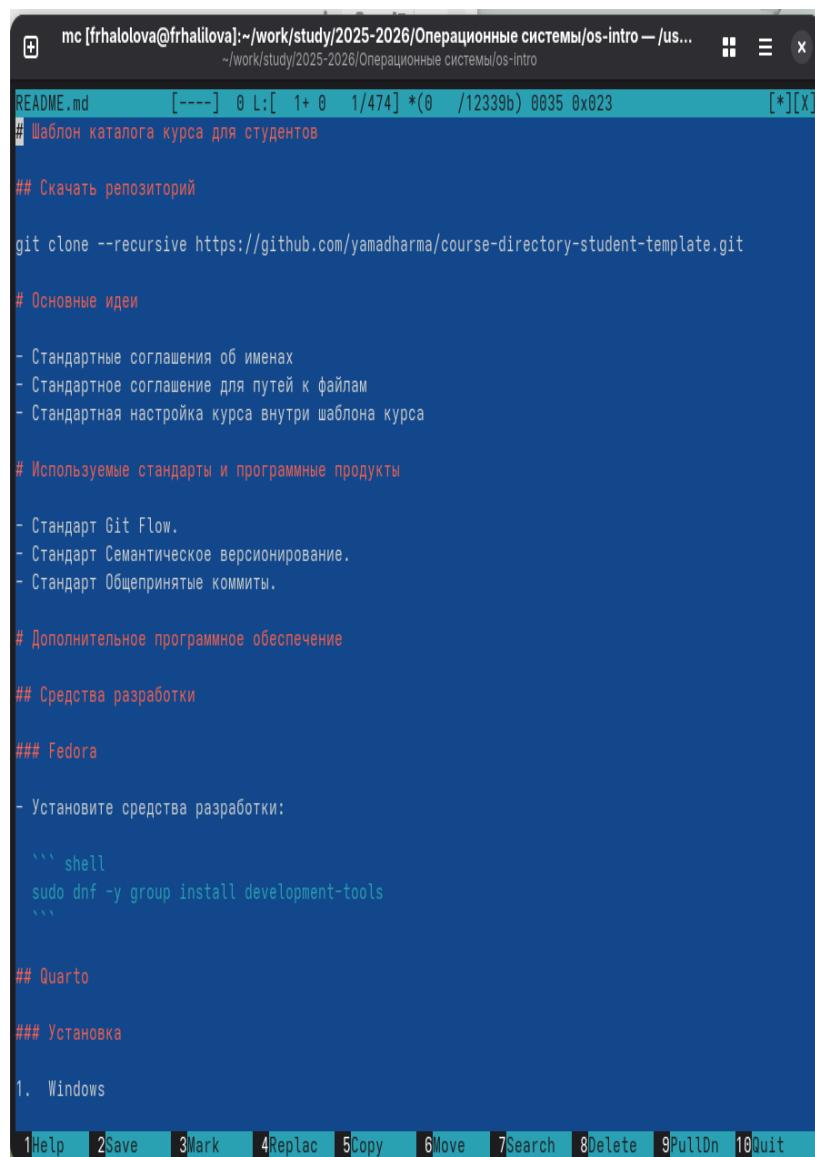


Рисунок 2.9: Дерево каталогов

5 Используя возможности подменю Файл , выполним:



The screenshot shows a terminal window with the following content:

```
mc [frhalilova@frhalilova]:~/work/study/2025-2026/Операционные системы/os-intro — /us... ~/work/study/2025-2026/Операционные системы/os-intro [*][X]
README.md      [----] 0 L:[ 1+ 0  1/474] *(0  /12339b) 0035 0x023
# Шаблон каталога курса для студентов

## Скачать репозиторий

git clone --recursive https://github.com/yamadharma/course-directory-student-template.git

# Основные идеи

- Стандартные соглашения об именах
- Стандартное соглашение для путей к файлам
- Стандартная настройка курса внутри шаблона курса

# Используемые стандарты и программные продукты

- Стандарт Git Flow.
- Стандарт Семантическое версионирование.
- Стандарт Общепринятые коммиты.

# Дополнительное программное обеспечение

## Средства разработки

### Fedora

- Установите средства разработки:
  ```` shell
 sudo dnf -y group install development-tools
  ````

## Quarto

### Установка

1. Windows
```

Рисунок 2.10: Просмотр содержимого текстового файла

The screenshot shows a terminal window titled 'mc [frhalolova@frhalolova:~/work/study/2025-2026/Операционные системы/os-intro — /us...]' with the path '/home/frhalolova/work/study/2025-2026/Операционные системы/os-intro/README.md'. The status bar indicates 1037/12339 lines and 8% completion. The file content includes:

```
# Шаблон каталога курса для студентов

## Скачать репозиторий

git clone --recursive https://github.com/yamadharma/course-directory-student-template.git

# Основные идеи

- Стандартные соглашения об именах
- Стандартное соглашение для путей к файлам
- Стандартная настройка курса внутри шаблона курса

# Используемые стандарты и программные продукты

- Стандарт Git Flow.
- Стандарт Семантическое версионирование.
- Стандарт Общепринятые коммиты.

# Дополнительное программное обеспечение

## Средства разработки

### Fedora

- Установите средства разработки:

```shell
sudo dnf -y group install development-tools
```

## Quarto

### Установка

1. Windows
```

The bottom of the window shows a menu bar with numbered options: 1 Help, 2 UnWrap, 3 Quit, 4 Hex, 5 Goto, 6, 7 Search, 8 Raw, 9 Format, 10 Quit.

Рисунок 2.11: Отредактируем содержимое текстового файла без сохранения результатов

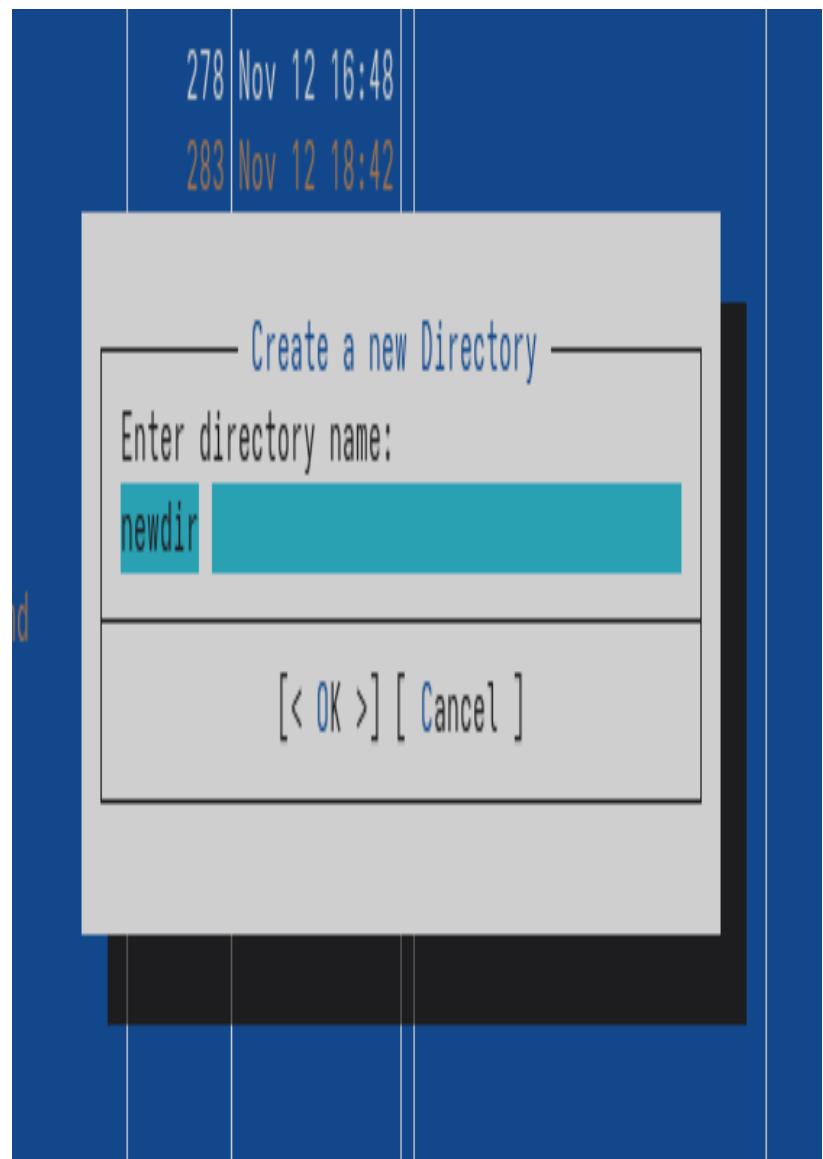


Рисунок 2.12: Создание каталога

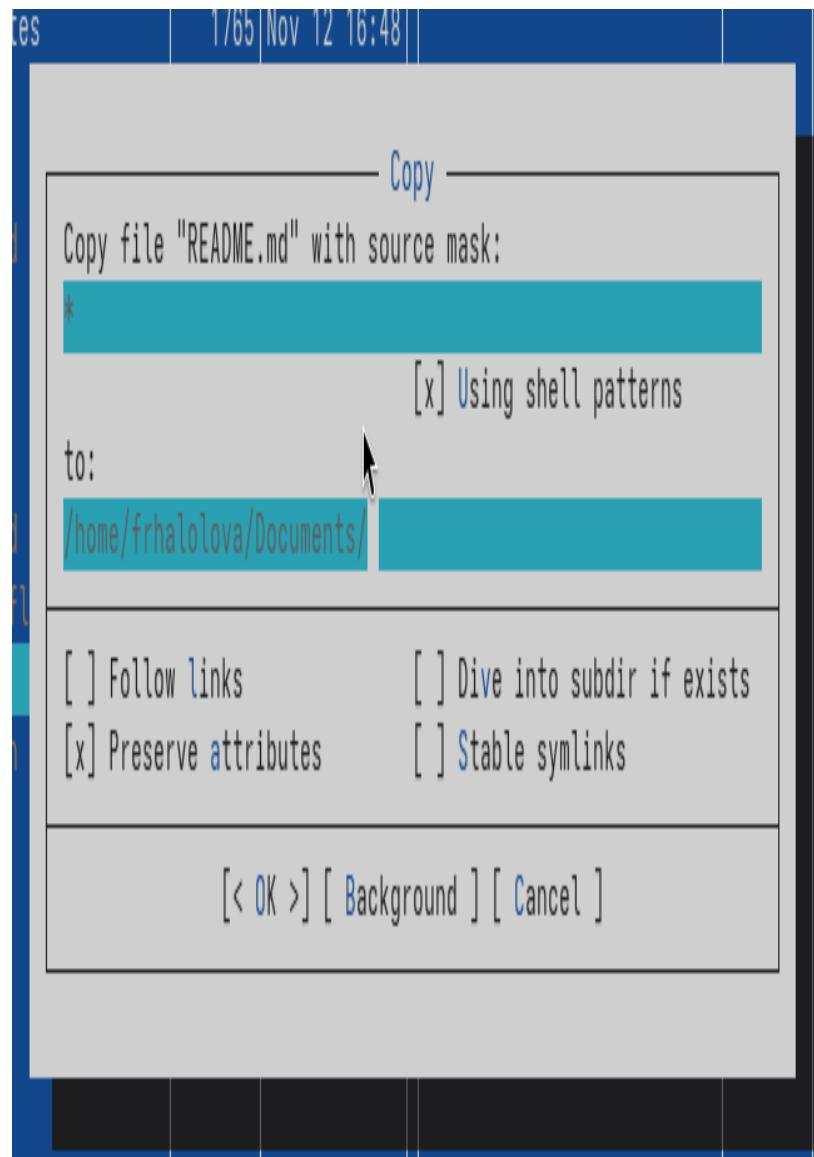


Рисунок 2.13: Копирование в файлов в созданный каталог

6. С помощью соответствующих средств подменю Команда осуществите:

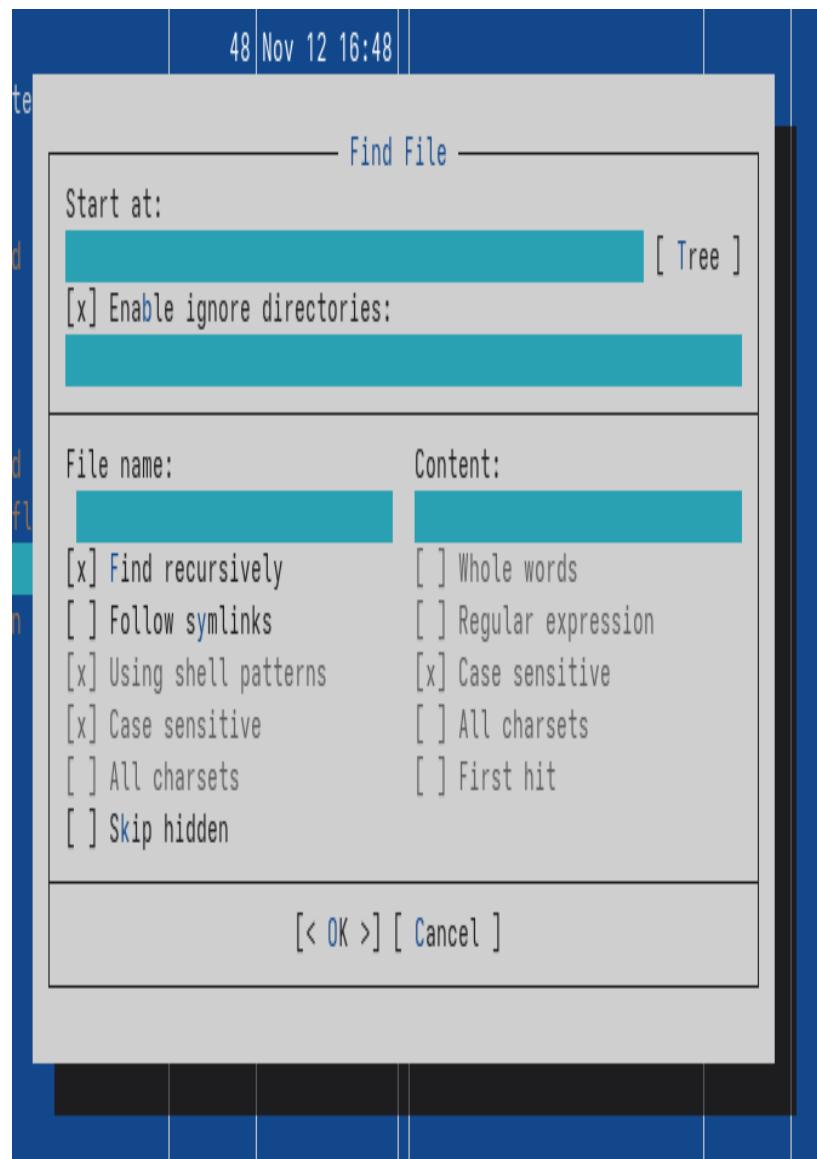


Рисунок 2.14: Поиск файлов

```
frhalolova@frhalilova:~$ ls -l
total 144
.bash_profile 144 Nov  8 2024
.bashrc 680 Nov 12 18:18
.gitconfig 232 Nov 12 18:43
abc1 0 Nov 13 19:07
conf.txt 760 Nov 13 19:25

UP--DIR History cd /newdir
ls 99G (82%)
Hint: If your terminal has keys, use the ESC+number sequence.

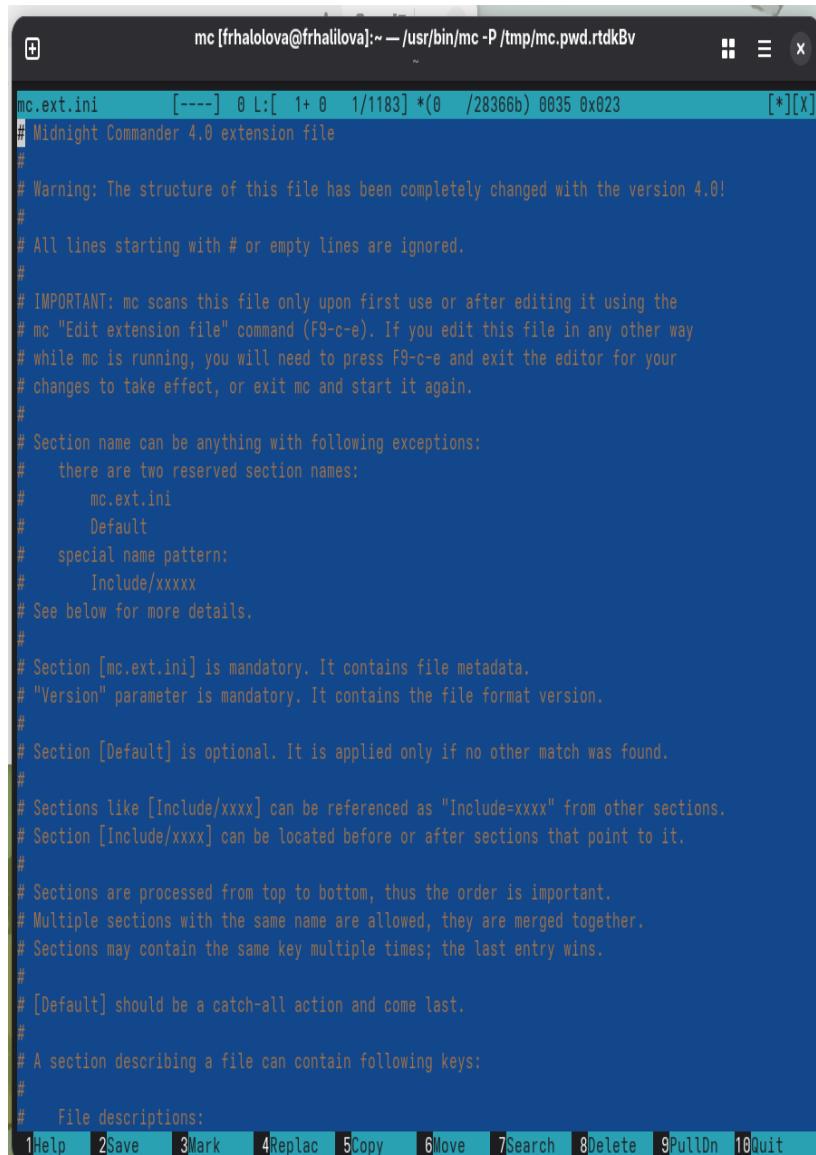
frhalolova@frhalilova:~$
```

1Help 2Menu 3View 4Edit 5Copy 6RenMov 7Mkdir 8Del

Рисунок 2.15: История команд



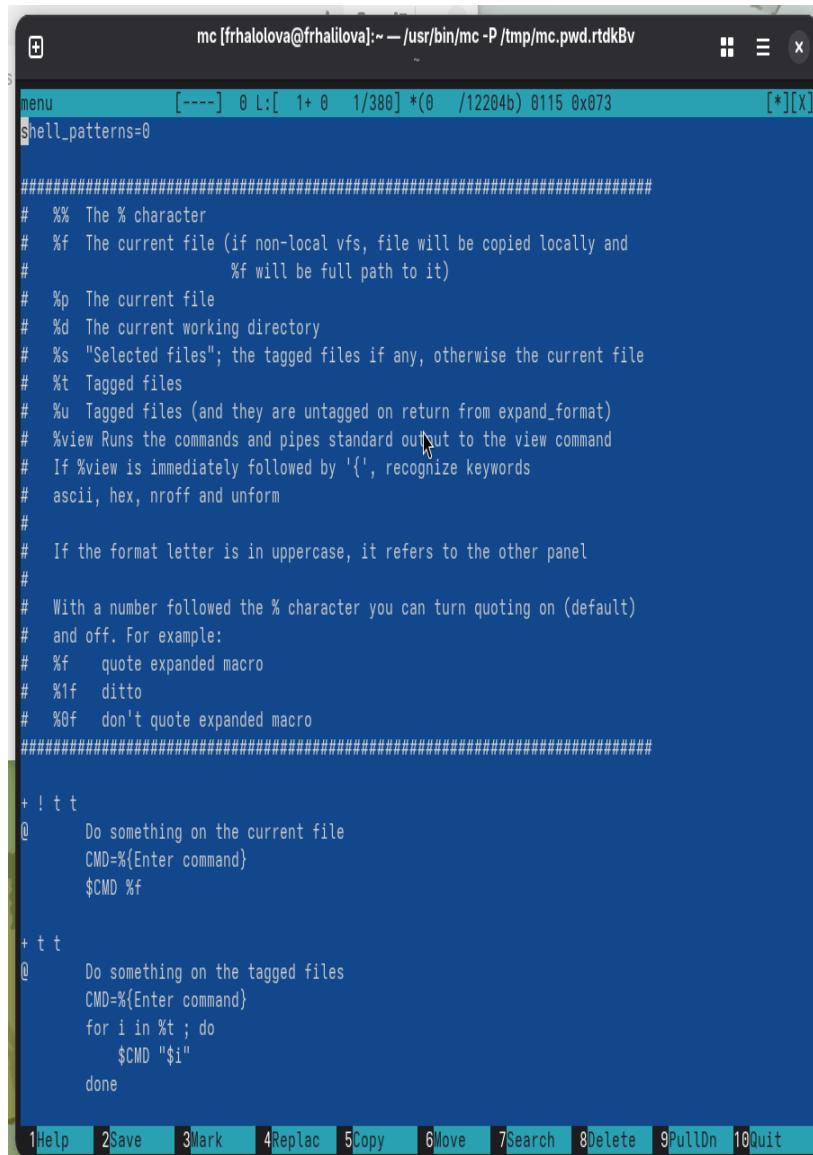
Рисунок 2.16: Переход в домашний каталог



mc [frhalolova@frhalilova]:~ — /usr/bin/mc -P /tmp/mc.pwd.rtdkBv

```
mc.ext.ini      [---]  0 L:[ 1+ 0  1/1183] *(0  /28366b) 0035 0x023      [*][X]
# Midnight Commander 4.0 extension file
#
# Warning: The structure of this file has been completely changed with the version 4.0!
#
# All lines starting with # or empty lines are ignored.
#
# IMPORTANT: mc scans this file only upon first use or after editing it using the
# mc "Edit extension file" command (F9-c-e). If you edit this file in any other way
# while mc is running, you will need to press F9-c-e and exit the editor for your
# changes to take effect, or exit mc and start it again.
#
# Section name can be anything with following exceptions:
#   there are two reserved section names:
#     mc.ext.ini
#     Default
#   special name pattern:
#     Include/xxxx
# See below for more details.
#
# Section [mc.ext.ini] is mandatory. It contains file metadata.
# "Version" parameter is mandatory. It contains the file format version.
#
# Section [Default] is optional. It is applied only if no other match was found.
#
# Sections like [Include/xxxx] can be referenced as "Include=xxxx" from other sections.
# Section [Include/xxxx] can be located before or after sections that point to it.
#
# Sections are processed from top to bottom, thus the order is important.
# Multiple sections with the same name are allowed, they are merged together.
# Sections may contain the same key multiple times; the last entry wins.
#
# [Default] should be a catch-all action and come last.
#
# A section describing a file can contain following keys:
#
#   File descriptions:
```

Рисунок 2.17: Просмотр файла расширений



The screenshot shows a terminal window with the title "mc [frhalolova@frhalilova]:~ — /usr/bin/mc -P /tmp/mc.pwd.rtdkBv". The window displays the content of the "menu" file. The file contains various shell patterns and command definitions for the mc command. The bottom of the window shows a menu bar with numbered options: 1Help, 2Save, 3Mark, 4Replac, 5Copy, 6Move, 7Search, 8Delete, 9PullDn, and 10Quit.

```
menu      [---]  0 L:[ 1+ 0  1/380] *(0  /12204b) 0115 0x073  [*][X]
shell.patterns=0

#####
# %% The % character
# %f The current file (if non-local vfs, file will be copied locally and
#       %f will be full path to it)
# %p The current file
# %d The current working directory
# %s "Selected files"; the tagged files if any, otherwise the current file
# %t Tagged files
# %u Tagged files (and they are untagged on return from expand_format)
# %view Runs the commands and pipes standard output to the view command
# If %view is immediately followed by '{', recognize keywords
# ascii, hex, nroff and unform
#
# If the format letter is in uppercase, it refers to the other panel
#
# With a number followed the % character you can turn quoting on (default)
# and off. For example:
# %f quote expanded macro
# %!f ditto
# %0f don't quote expanded macro
#####

+ ! t t
@    Do something on the current file
CMD=%{Enter command}
$CMD %f

+ t t
@    Do something on the tagged files
CMD=%{Enter command}
for i in %t ; do
    $CMD "$i"
done
```

Рисунок 2.18: Просмотр файла меню

7. Вызовем подменю Настройки. Изучим опции

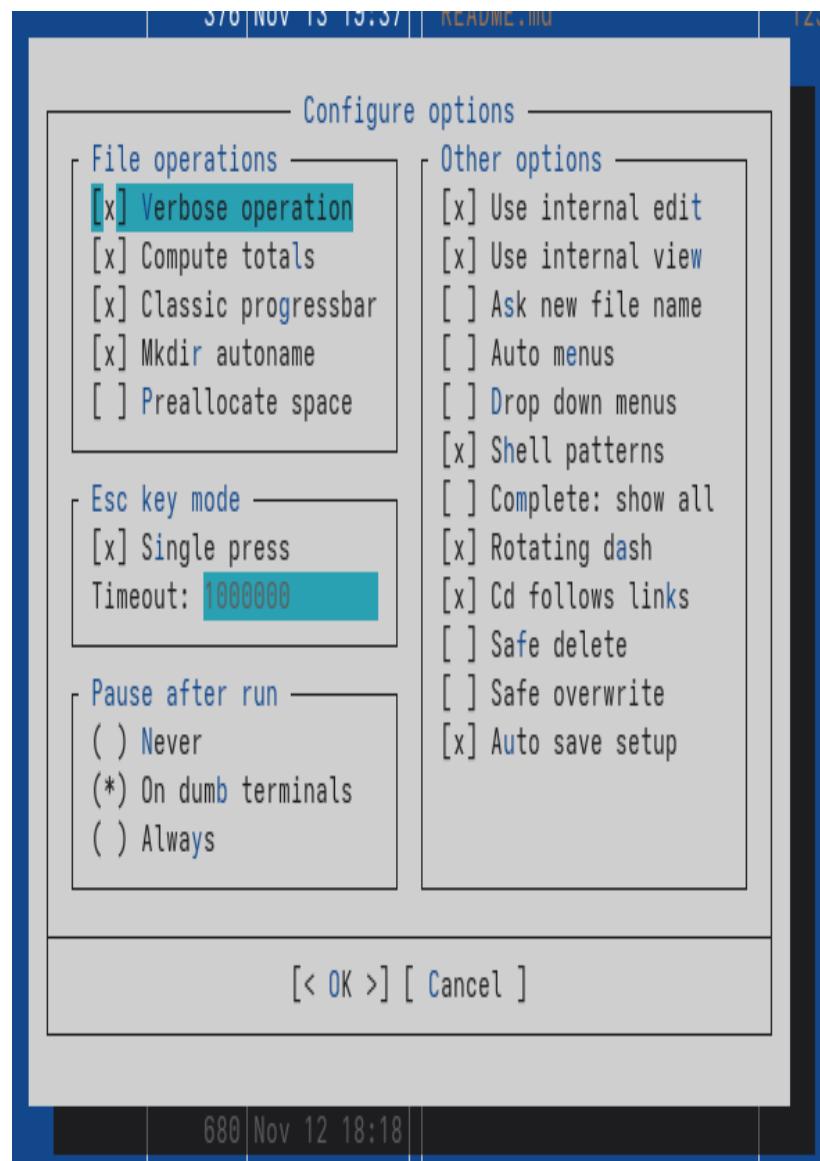


Рисунок 2.19: Конфигурация

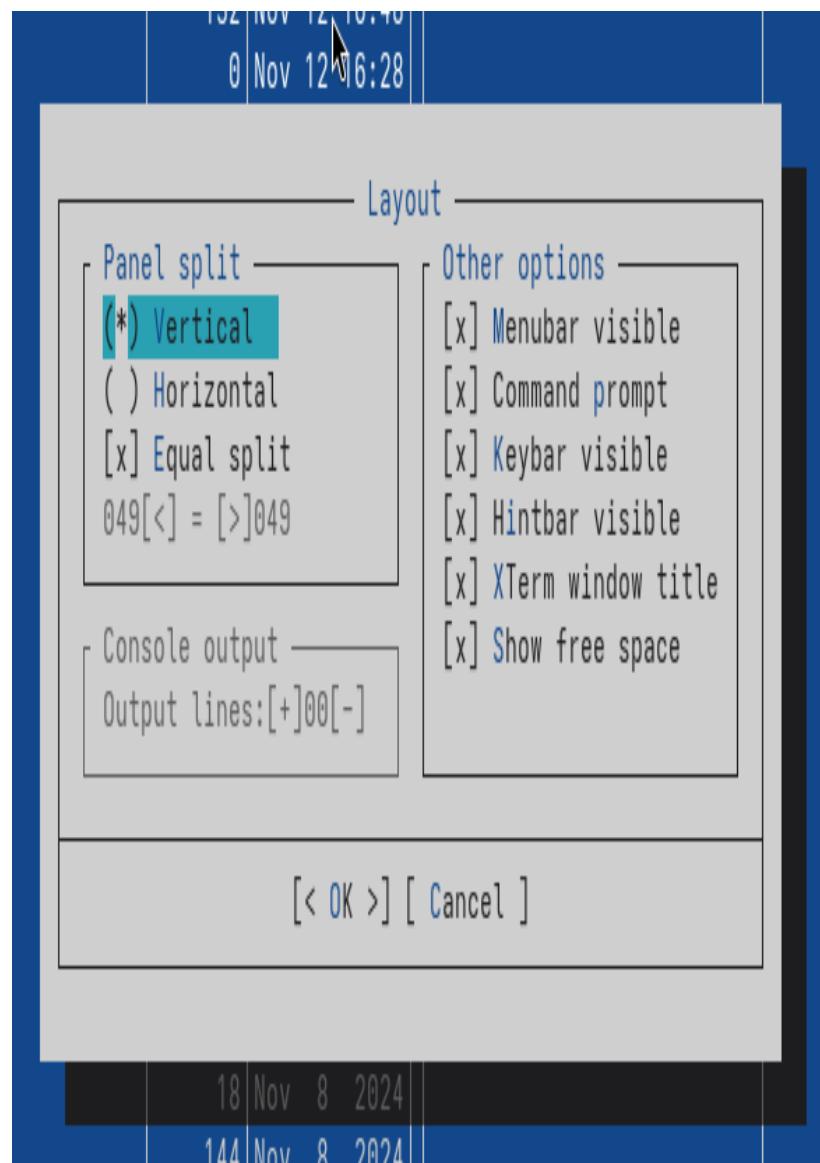


Рисунок 2.20: Внешний вид

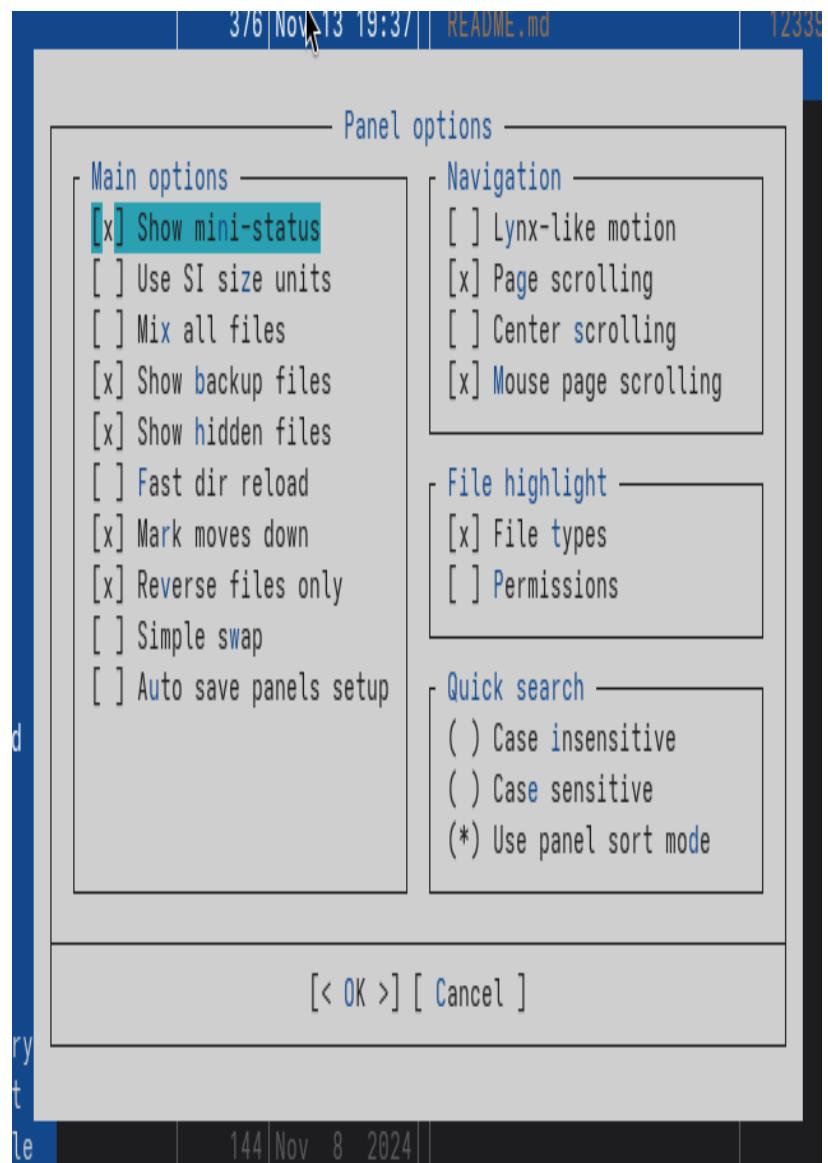


Рисунок 2.21: Настройки панелей

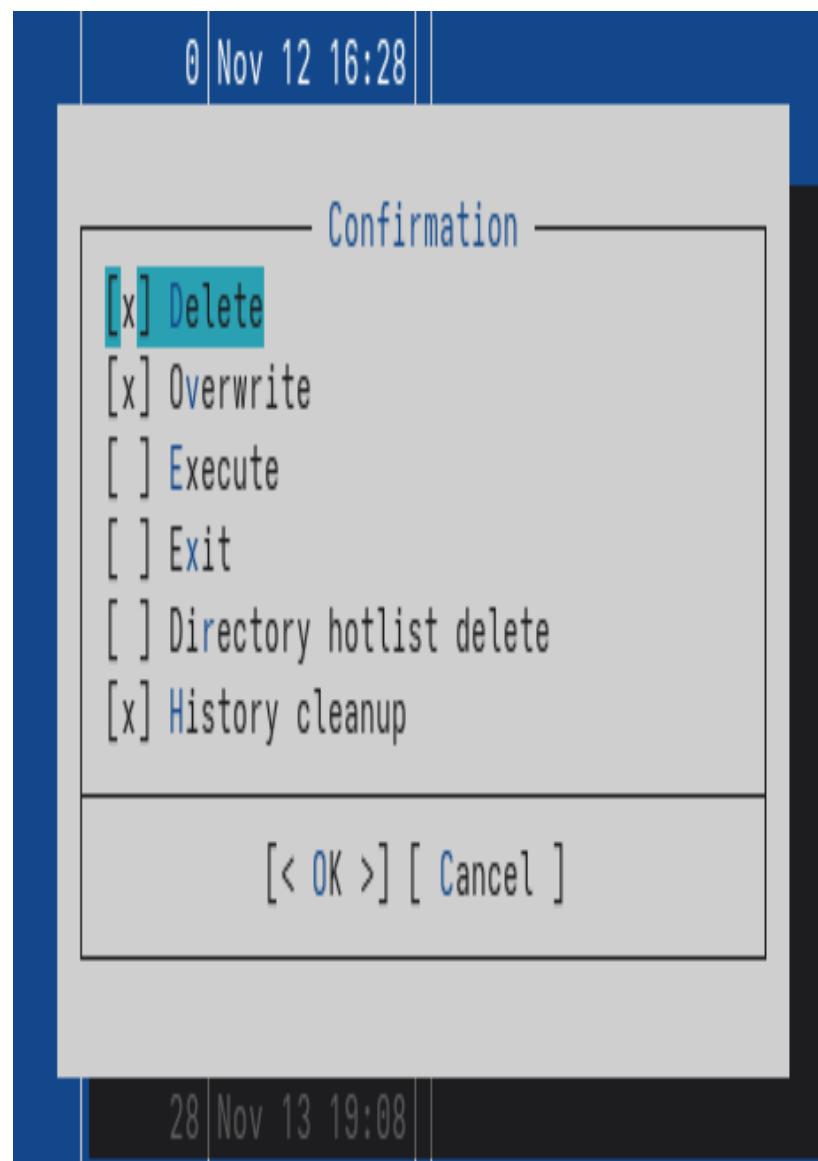


Рисунок 2.22: Подтверждение

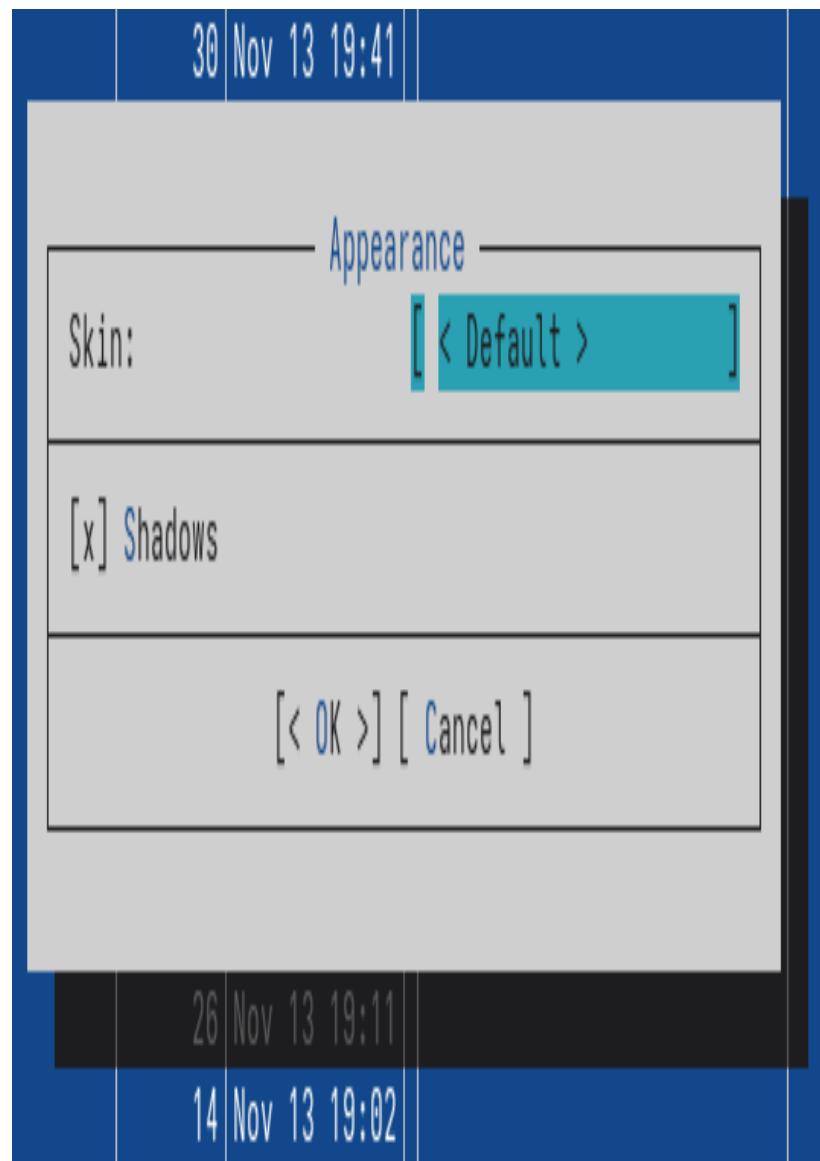


Рисунок 2.23: Оформление

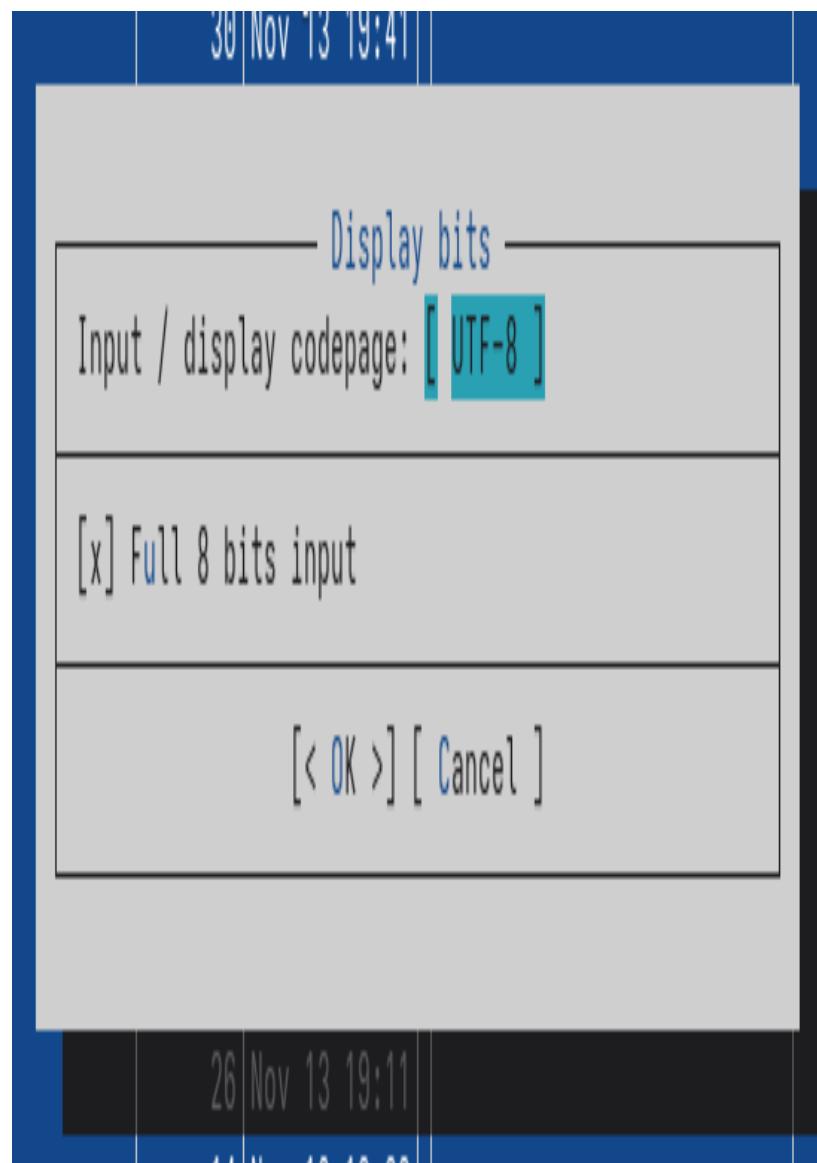


Рисунок 2.24: Кодировка символов

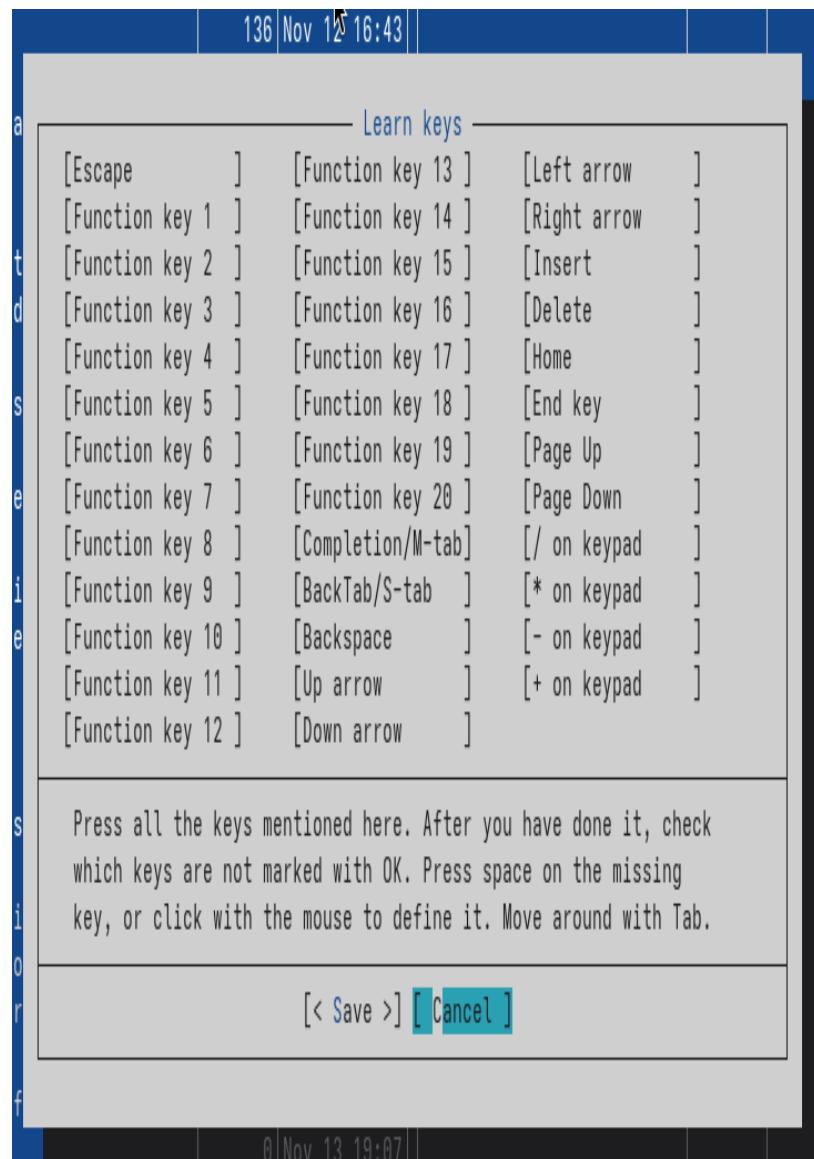


Рисунок 2.25: Распознавание клавиш

8 Создадим текстовой файл text.txt.

9 Откроем этот файл с помощью встроенного в ms редактора, и вставим в открытый файл небольшой фрагмент текста, скопированный из любого другого файла или Интернета.

```
mc [frhalilova@frhalilova:~/Documents -- /usr/bin/mc -P /tmp/mc.pwd.rtdkBv
~/Documents
README.md      [---]  0 L:[ 1+ 0  1/474] *(0  /12339b) 0035 0x023
[*][X]

# Шаблон каталога курса для студентов

## Скачать репозиторий

git clone --recursive https://github.com/yamadharma/course-directory-student-template.git

# Основные идеи

- Стандартные соглашения об именах
- Стандартное соглашение для путей к файлам
- Стандартная настройка курса внутри шаблона курса

# Используемые стандарты и программные продукты

- Стандарт Git Flow.
- Стандарт Семантическое версионирование.
- Стандарт Общепринятые коммиты.

# Дополнительное программное обеспечение

## Средства разработки

### Fedora

- Установите средства разработки:

  ```` shell
 sudo dnf -y group install development-tools
  ````

## Quarto

### Установка

1. Windows
```

Рисунок 2.26: Файл с текстом

10 Проделаем с текстом следующие манипуляции, используя горячие клавиши:

Удалим строку текста. - F8

The screenshot shows a terminal window titled "mc [frhalilova@frhalilova]:~/Documents — /usr/bin/mc -P /tmp/mc.pwd.rtdkBv". The current directory is "/Documents". The file "README.md" is open, displaying the following content:

```
README.md      [-M--]  0 L:[ 1+12 13/452] *(471 /11786b) 0049 0x031      [*][X]
# Шаблон каталога курса для студентов

## Скачать репозиторий

git clone --recursive https://github.com/yamadharma/course-directory-student-template.git

# Основные идеи

- Стандартные соглашения об именах
- Стандартное соглашение для путей к файлам
- Стандартная настройка курса внутри шаблона курса

1. Windows

- Chocolatey (см. [Пакетный менеджер для Windows.
Chocolatey](id:78f49355-a68d-4cb0-9f35-e7abd8b180df)):

    ```` shell
 choco install quarto
    ````

2. Linux

1. Linux в общем

- Установка с помощью скрипта:

    ```` shell
 #!/bin/bash

 ## Система
 TARGET=/opt
 TARGET_BIN=/usr/local/bin
 ## Домашний каталог
 # TARGET=~/.opt
 # TARGET_BIN=~/.local/bin
```

Рисунок 2.27: Файл с текстом

Выделите фрагмент текста и скопируйте его на новую строку. - F5

```
mc [frhalilova@frhalilova]:~/Documents — /usr/bin/mc -P /tmp/mc.pwd.rtdkBv
~/Documents [*][X]

README.md [BM--] 0 L:[9+35 44/452] *(1233/11786b) 0032 0x020

- Стандартные соглашения об именах
- Стандартное соглашение для путей к файлам
- Стандартная настройка курсора внутри шаблона курса

1. Windows

- Chocolatey (см. [Пакетный менеджер для Windows].
Chocolatey](id:78f49355-a60d-4cb0-9f35-e7abd8b180df)):

``` shell
choco install quarto
```

2. Linux

1. Linux в общем

- Установка с помощью скрипта:

``` shell
#!/bin/bash

## Система
TARGET=/opt
TARGET_BIN=/usr/local/bin
## Домашний каталог
# TARGET=~/.opt
# TARGET_BIN=~/.local/bin

## Получить тег
TAG=`basename $(curl -sL -o /dev/null -w %{url_effective} https://github.com/quarto-dev/
TAG=${TAG/v/}`

## Скачать
cd /tmp
```

Рисунок 2.28: Копирование фрагмента

Сохраним файл. - F2

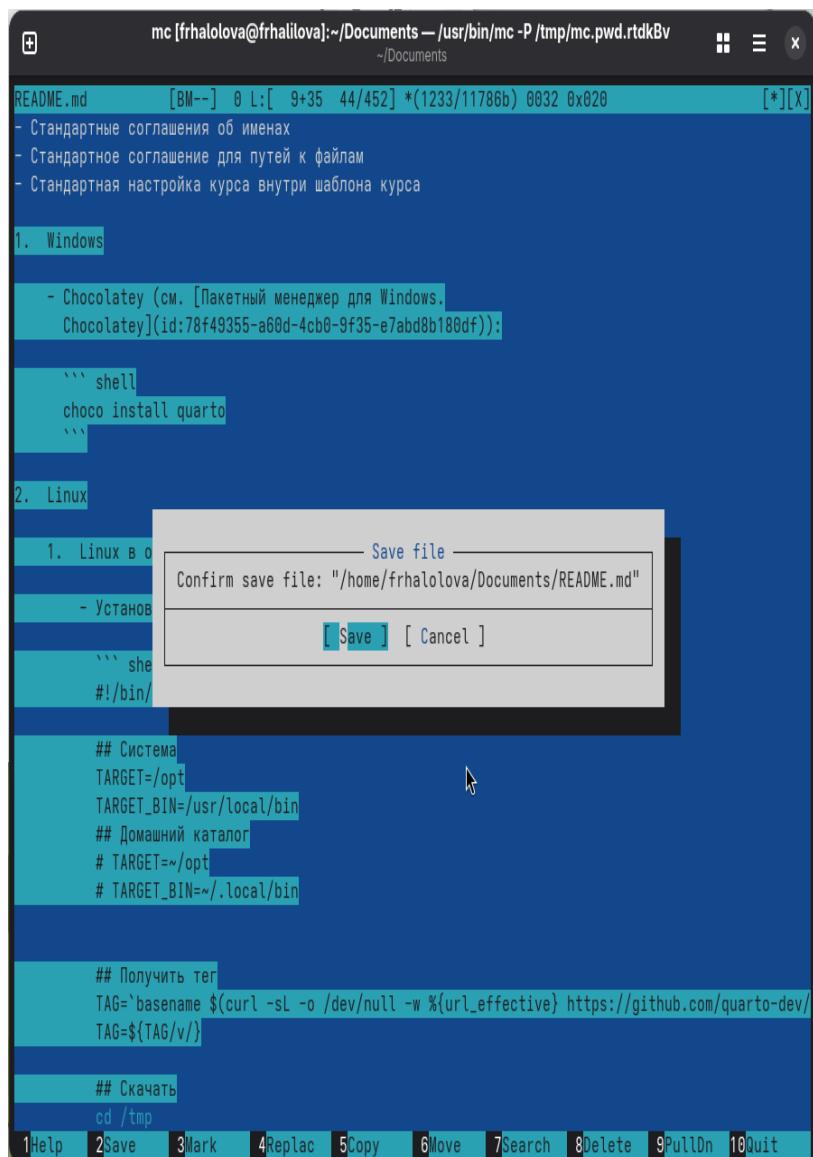


Рисунок 2.29: Сохранение

Отменим последнее действие. - Ctrl+U

The screenshot shows a terminal window with the following content:

```
mc [frhalolova@frhalolova]:~/Documents — /usr/bin/mc -P /tmp/mc.pwd.rtdkBv ~/Documents
```

File: README.md

```
[B[—] 0 L:[ 9+16 25/452] *(714 /11810b) 0010 0x00A [*][X]
```

- Стандартные соглашения об именах
- Стандартное соглашение для путей к файлам
- Стандартная настройка курса внутри шаблона курса

1. Windows

```
- Chocolatey (см. [Пакетный менеджер для Windows.  
Chocolatey](id:78f49355-a60d-4cb0-9f35-e7abd8b180df)):
```

```
``` shell  
choco install quarto
```
```

2. Linux

1. Linux в общем

```
- Установка с помощью скрипта:
```

```
``` shell  
#!/bin/bash

Система
TARGET=/opt
TARGET_BIN=/usr/local/bin
Домашний каталог
TARGET=~/.opt
TARGET_BIN=~/.local/bin

Получить тег
TAG=`basename $(curl -sL -o /dev/null -w %{url_effective} https://github.com/quarto-dev/quarto-dev)
TAG=${TAG/v/}`

Скачать
u;50;21M;50;21m;33;16M0Acd /tmp
```

1 Help 2 Save 3 Mark 4 Replace 5 Copy 6 Move 7 Search 8 Delete 9 PullDown 10 Quit

Рисунок 2.30: Отмена

Перейдем в конец файла - PageDown или Ctrl+X

The screenshot shows a terminal window titled 'mc [frhalilova@frhalilova]:~/Documents — /usr/bin/mc -P /tmp/mc.pwd.rtdkBv'. The window displays a shell script for GitHub release automation. The script includes commands for pushing changes to GitHub, creating a release, and publishing a video. The terminal interface includes a status bar at the bottom with numbered menu options from 1 to 10.

```
README.md [BM--] 0 L:[417+35 452/452] *(11810/11810b) <EOF> [*][X]
``` shell
git flow release finish 1.0.0
```

- Отправим данные на github

``` shell
git push --all
git push --tags
```

- Скопируем CHANGELOG.md в каталог `release`:

``` shell
mkdir -p ./release
cp CHANGELOG.md ./release
```

- Создадим релиз на github. Для этого будем использовать утилиты работы с github (см. [github: утилиты командной строки](id:d1925a41-6b4c-4a3a-b182-6337891b8841)):

``` shell
gh release create v1.0.0 -F ./release/CHANGELOG.md
```

Видео

[![RuTube](https://pic.rutubelist.ru/video/2025-09-03/f3/71/f371923b6802db01d0bc4735586e7b95.jpg)]
[![Платформа](https://s-dt-rt1.cloud.edgecore.ru/fp-2025-09-cover/Jjf4mXm-h65_607985b3-64d7-416f-)]
[![VKvideo](https://sun1-86.userapi.com/X4ozpnQbDz0szkmLpm2NpauUx075QbpjA6fVGg/p0VZsIv3Ees.jpg)]
[![Youtube](http://img.youtube.com/vi/39bu5avPoDU/0.jpg)](http://www.youtube.com/watch?v=39bu5avPoDU)

1Help 2Save 3Mark 4Replac 5Copy 6Move 7Search 8Delete 9PullDn 10Quit
```

Рисунок 2.31: Переход в конец файла

Перейдем в начало файла - PageUp или Ctrl+Z

The screenshot shows a terminal window titled 'mc [frhalilova@frhalilova:~/Documents -- /usr/bin/mc -P /tmp/mc.pwd.rtdkBv ~/Documents [\*][X]'. The current directory is ~/Documents. The file 'README.md' is open, displaying its contents. The text in 'README.md' includes instructions for cloning a repository, listing standard conventions, and providing setup steps for Windows and Linux. The bottom of the window shows a menu bar with numbered options: 1Help, 2Save, 3Mark, 4Replace, 5Copy, 6Move, 7Search, 8Delete, 9PullDown, and 10Quit.

```
mc [frhalilova@frhalilova:~/Documents -- /usr/bin/mc -P /tmp/mc.pwd.rtdkBv ~/Documents [*][X]
README.md [B[--] 35 L:[1+ 0 1/452] *(63 /11811b) 1090 0x442
Шаблон каталога курса для студентов

Скачать репозиторий

git clone --recursive https://github.com/yamadharma/course-directory-student-template.git

Основные идеи

- Стандартные соглашения об именах
- Стандартное соглашение для путей к файлам
- Стандартная настройка курса внутри шаблона курса

1. Windows

- Chocolatey (см. [Пакетный менеджер для Windows.
Chocolatey](id:78f49355-a60d-4cb0-9f35-e7abd8b180df)):

``` shell
choco install quarto
```

2. Linux

1. Linux в общем

- Установка с помощью скрипта:

``` shell
#!/bin/bash

## Система
TARGET=/opt
TARGET_BIN=/usr/local/bin
## Домашний каталог
# TARGET=~/.opt
# TARGET_BIN=~/.local/bin
```
1Help 2Save 3Mark 4Replace 5Copy 6Move 7Search 8Delete 9PullDown 10Quit
```

Рисунок 2.32: Переход в начало файла

11 Откроем файл с исходным текстом на языке программирования C

```
tcp_sk_buf_3_10_to_4_8.c [----] 0 L:[13+ 0 13/ 84] *(81 /1588b) 0115 0x073 [*][X]
static void
tcp_showusage()
{
<-----> printf("usage : tcp %s", tcp_usage());
}

string
tcp_help()
{
<-----> return "Help";
}

int
tcp()
{
<----->int i;
<----->int size;
<----->struct tcp_table *table;
<----->struct sock_common *off = 0;

<----->table = (struct tcp_table *)&tcp_table;

<----->for (i = 0; i < table->mask; i++) {
<-----><----->struct hlist_nulls_node *pos;

<-----><----->pos = table->hash[i].head.first;

<-----><----->while (!(((unsigned long)pos & 1)) {
<-----><-----><----->struct sock *sk;
<-----><-----><----->struct sk_buff *next;
<-----><-----><----->struct sk_buff_head *head;
<-----><-----><----->struct hlist_nulls_node *node;

<-----><-----><----->sk = (struct sock *)(((unsigned long)pos - ((unsigned long)&(off->skc_dont

<-----><-----><----->head = (struct sk_buff_head *)&(sk->sk_receive_queue);
1Help 2Save 3Mark 4Replace 5Copy 6Move 7Search 8Delete 9PullDown 10Quit
```

Рисунок 2.33: Файл с программой

12 Используя меню редактора, выключим подсветку синтаксиса.

The screenshot shows a terminal window titled 'mc [frhalolova@frhalilova]:~ — /usr/bin/mc -P /tmp/mc.pwd.ceAaBG'. The command line below it is 'frhalolova@frhalilova:~/Documents'. The terminal displays a portion of a C program named 'udp\_sk\_buf\_3\_18\_to\_4\_8.c'. A context menu is open at the bottom of the code area, specifically over the 'C Program' option in the 'Choose syntax highlighting' submenu. The submenu also includes options like 'Auto', 'Reload Current Syntax', and other programming language choices. At the bottom of the terminal window, there is a row of numbered buttons from 1 to 10, each corresponding to a different command: Help, Save, Mark, Replace, Copy, Move, Search, Delete, PullDown, and Quit.

```

static void
udp_showusage()
{
<-----> printf("usage : udp %s", udp_usage());
}

string
udp_help()
{
<-----> return "Help";
}

int
udp()
{
<----->int i;
<----->int size;
<----->struct udp_table *t
<----->struct sock_common
<----->table = (struct udp
<----->for (i = 0; i < tab
<----->struct hlis
<-----><----->pos = table->hash[i].head.first;

<-----><----->while (!((unsigned long)pos & 1)) {
<-----><-----> struct sock *sk;
<-----><-----> struct sk_buff *next;
<-----><-----> struct sk_buff_head *head;
<-----><-----> struct hlist_nulls_node *node;

<-----><-----> sk = (struct sock *)((unsigned long)pos - ((unsigned long)&(off->skc_dont
<-----><-----> head = (struct sk_buff_head *)&(sk->sk_receive_queue);

```

1 Help 2 Save 3 Mark 4 Replace 5 Copy 6 Move 7 Search 8 Delete 9 PullDown 10 Quit

Рисунок 2.34: Цветовыделение синтаксиса

## **3 Вывод**

В данной работе мы ознакомились с инструментами командной оболочки Midnight Commander. Приобрели навыки практической работы по просмотру каталогов и файлов

## 4 Контрольные вопросы

1. Какие режимы работы есть в mc? Охарактеризуйте их. Ответ: В командной оболочке mc есть два режима Информация и Дерево. В режиме Информация на панель выводятся сведения о файле и текущей файловой системе, расположенных на активной панели. В режиме Дерево на одной из панелей выводится структура дерева каталогов. Управлять режимами отображения панелей можно через пункты меню mc
2. Какие операции с файлами можно выполнить как с помощью команд shell, так и с помощью меню mc? Привести несколько примеров. Ответ: Командные интерпретатор Shell и оболочки Midnight Commander имеют похожую структуру и многие одинаковые команды можно выполнить в обоих оболочках вот некоторые из них
  - a) Системная информация
  - b) Поиск
  - c) Копирование
3. Опишите структуру меню левой панели mc, дайте характеристику командам. Ответ: Меню левой панели mc представляет собой следующую конструкцию:

Где подпункты меню a) Список файлов показывает файлы в домашнем каталоге. b) Быстрый просмотр позволяет выполнить быстрый просмотр содержимого панели.

с) Информация позволяет посмотреть информацию о файле или каталоге d) Командная оболочка Midnight Commander В меню каждой (левой или правой) панели можно выбрать Формат списка: стандартный, ускоренный, расширенный и определённый пользователем. е) Порядок сортировки позволяет задать критерии сортировки при выводе списка файлов и каталогов: без сортировки, по имени, расширенный, время правки, время доступа, время изменения атрибута, размер, узел.

4. Опишите структура меню Файл mc и дайте характеристику командам.

Ответ: Меню Файл mc представляет собой следующую конструкцию:

Где подпункты меню а) Просмотр ( F3 ) позволяет посмотреть содержимое текущего файла без возможности редактирования. б) – Просмотр вывода команды ( M + ! ) функция запроса команды с параметрами. с) Правка ( F4 ) открывает текущий (или выделенный) файл для его редактирования. д) Копирование ( F5 ) осуществляет копирование одного или нескольких файлов или каталогов в указанное пользователем во всплывающем окне место. е) Права доступа ( Ctrl-x с ) позволяет изменить права доступа к одному или нескольким файлам или каталогам. ф) Права доступа на файлы и каталоги г) Жёсткая ссылка ( Ctrl-x 1 ) позволяет создать жёсткую ссылку к текущему (или выделенному) файлу1 . х) Символическая ссылка ( Ctrl-x s ) – позволяет создать символическую ссылку к текущему файлу . и) Владелец группы ( Ctrl-x о ) позволяет задать владельца и имя группы для одного или нескольких файлов или каталогов. ж) Права (расширенные) позволяет изменить права доступа и владения для одного или нескольких файлов или каталогов. к) Переименование ( F6 ) позволяет переименовать один или несколько файлов или каталогов. л) Создание каталога ( F7 ) позволяет создать каталог. м) Удалить ( F8 ) позволяет удалить один или несколько файлов или каталогов. н) Выход ( F10 ) завершает работу mc.

5 Опишите структура меню Команда mc, дайте характеристику командам

Ответ: Ответ: Меню Команда mc представляет собой следующую конструкцию:

Где подпункты меню a) Дерево каталогов отображает структуру каталогов системы. b) Поиск файла выполняет поиск файлов по заданным параметрам. c) Переставить панели меняет местами левую и правую панели. d) Сравнить каталоги (Ctrl-x d) сравнивает содержимое двух каталогов. e) Размеры каталогов отображает размер и время изменения каталога (по умолчанию в mc размер каталога корректно не отображается). f) История командной строки выводит на экран список ранее выполненных в оболочке команд. g) Каталоги быстрого доступа (Ctrl-) при вызове выполняется быстрая смена текущего каталога на один из заданного списка. h) Восстановление файлов позволяет восстановить файлы на файловых системах ext2 и ext3. i) Редактировать файл расширений позволяет задать с помощью определённого синтаксиса действия при запуске файлов с определённым расширением (например, какое программное обеспечение запускать для открытия или редактирования файлов с расширением .c или .cpp). j) Редактировать файл меню позволяет отредактировать контекстное меню пользователя, вызываемое по клавише F2 . k) Редактировать файл расцветки имён позволяет подобрать оптимальную для пользователя расцветку имён файлов в зависимости от их типа.

6. Опишите структура меню Настройки mc, дайте характеристику командам  
Ответ: Меню Настройки mc представляет собой следующую конструкцию:

Где подпункты меню a) Конфигурация позволяет скорректировать настройки работы с панелями. b) Внешний вид и Настройки панелей определяет элементы, отображаемые при вызове mc, а также цветовое выделение. c) Биты символов задаёт формат обработки информации локальным терминалом. d) Подтверждение позволяет установить или убрать вывод окна с запросом подтверждения действий при операциях удаления и перезаписи файлов, а также

при выходе из программы. e) Распознание клавиш диалоговое окно используется для тестирования функциональных клавиш, клавиш управления курсором и прочее. f) Виртуальные ФС настройки виртуальной файловой системы: тайм-аут, пароль и прочее.

7. Назовите и дайте характеристику встроенным командам тс. Ответ: В командную оболочку тс встроены стандартные команды. Вот некоторые из них.

- a) F1 Вызов контекстно-зависимой подсказки.
- b) F2 Вызов пользовательского меню с возможностью создания and/or.
- c) F3 Просмотр содержимого файла, на который указывает подсветка в активной панели.
- d) F4 Вызов встроенного в тс редактора для изменения содержания файла, на который указывает подсветка в активной панели.
- e) F5 Копирование одного или нескольких файлов, отмеченных в первой (активной) панели, в каталог, отображаемый на второй панели.
- f) F6 Перенос одного или нескольких файлов, отмеченных в первой панели, в каталог, отображаемый на второй панели.
- g) F7 Создание подкатаога в каталоге, отображаемом в активной панели.
- h) F8 Удаление одного или нескольких файлов, отмеченных в первой панели файлов.
- i) Вызов меню тс.
- j) F10 Выход из тс.

8 Назовите и дайте характеристику командам встроенного редактора тс. Ответ: В редактор тс встроено немало команд. Вот некоторые из них. a) Ctrl+у удалить строку. b) Ctrl+и отмена последней операции. c) Ins вставка/замена. d) F7 поиск. d) Shift+F7 повтор последней операции поиска. e) F4 замена файла. f) F3 первое нажатие начало выделения, второе это окончание выделения. g)

F5 копировать выделенный фрагмент F6 переместить выделенный фрагмент.  
h) F8 удалить выделенный фрагмент. i) F2 записать изменения в файл. j) F10  
выйти из редактора.

9. Дайте характеристику средствам mc, которые позволяют создавать меню, определяемые пользователем. Ответ: Один из четырех форматов списка в Midnight Commander -Пользовательский определённый самим пользователем позволяет ему редактировать меню любого из двух списков. А меню пользователя – это меню, состоящее из команд, определенных пользователем. При вызове меню используется файл `~/.mc.menu`. Если такого файла нет, то по умолчанию используется системный файл меню `/usr/lib/mc/mc.menu`. Все строки в этих файлах , начинающиеся с пробела или табуляции, являются командами, которые выполняются при выборе записи.
10. Дайте характеристику средствам mc, которые позволяют выполнять действия, определяемые пользователем, над текущим файлом Ответ: Когда мы выделяем файл не являющегося исполняемым, Midnight Commander сравнивает расширение выбранного файла с расширениями, прописанными в «файле расширений» `~/mc.ext`. Если в файле расширений найдется подраздел, задающий процедуры обработки файлов с данным расширением, то обработка файла производится в соответствии с заданными в этом подразделе командами и файлами:
  - a) файл помощи для МС. `/usr/lib/mc.hlp`
  - b) файл расширений, используемый по умолчанию. `/usr/lib/mc/mc.ext`
  - c) файл расширений, конфигурации редактора. `$HOME/.mc.ext`
  - d) системный инициализационный файл. `/usr/lib/mc/mc.ini`
  - e) файл который содержит основные установки. `/usr/lib/mc/mc.lib`
  - f) инициализационный файл пользователя. Если он существует, то системный файл `mc.ini` игнорируется. `$HOME/.mc.ini`

g) этот файл содержит подсказки, отображаемые в нижней части экрана.

/usr/lib/mc/mc.hint

h) системный файл меню MC, используемый по умолчанию. /usr/lib/mc/mc.menu

i) файл меню пользователя. Если он существует, то системный файл меню игнорируется. \$HOME/.mc.menu

j) инициализационный файл пользователя. Если он существует, то системный файл mc.ini игнорируется. \$HOME/.mc.tree