

# Командная оболочка Midnight Commander

---

Ларина Наталья<sup>1</sup>

27 марта, 2024, Москва, Россия

<sup>1</sup>Российский Университет Дружбы Народов

# Цели и задачи работы

---

## Цель лабораторной работы

Ознакомление с файловой системой Linux, её структурой, именами и содержанием каталогов. Приобретение практических навыков по применению команд для работы с файлами и каталогами, по управлению процессами, по проверке использования диска и обслуживанию файловой системы.

# Задачи лабораторной работы

- 1 Изучить возможности Midnight Commander
- 2 Изучить редактор Midnight Commander

# **Процесс выполнения лабораторной работы**

---

# Работа с Midnight Commander

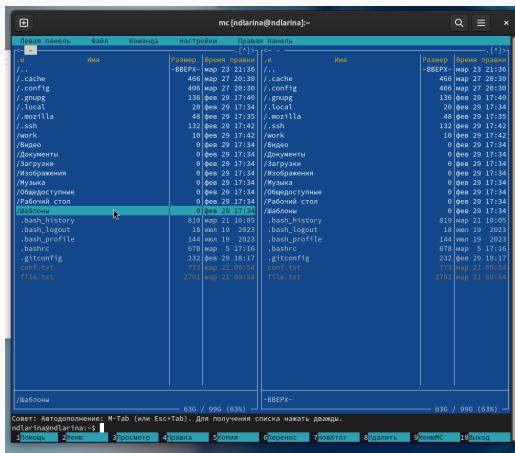


Рис. 1: Запуск mc

# Работа с Midnight Commander

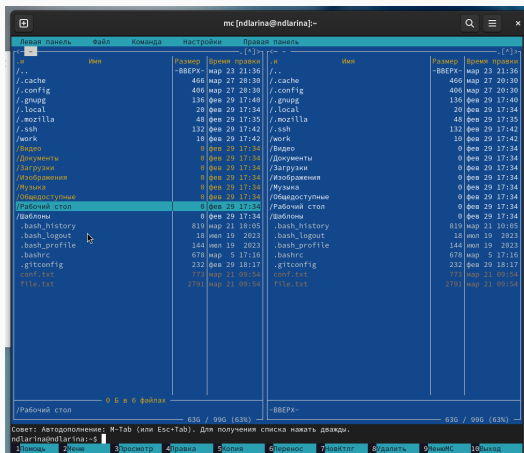


Рис. 2: Выделение

# Работа с Midnight Commander

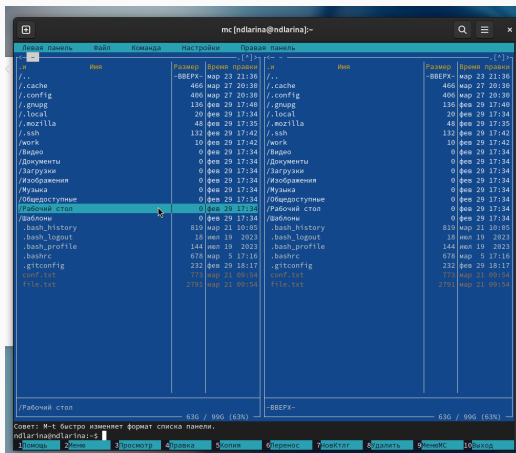


Рис. 3: Отмена



# Работа с Midnight Commander

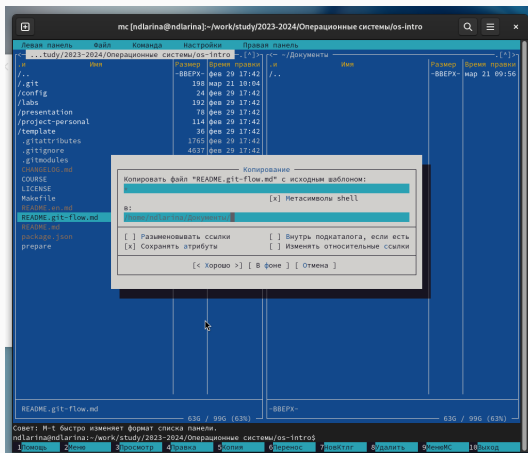


Рис. 4: Копирование

# Работа с Midnight Commander

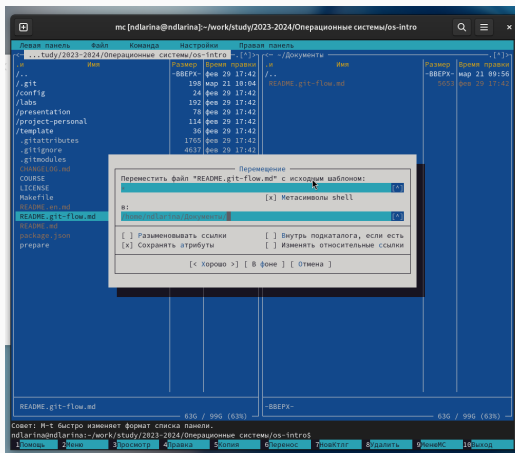


Рис. 5: Перемещение

# Работа с Midnight Commander

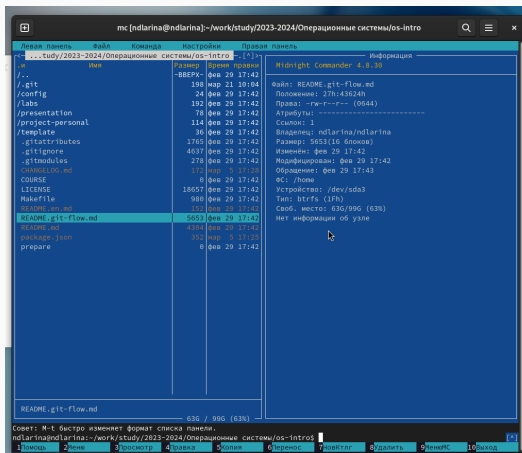


Рис. 6: Информация

# Работа с Midnight Commander

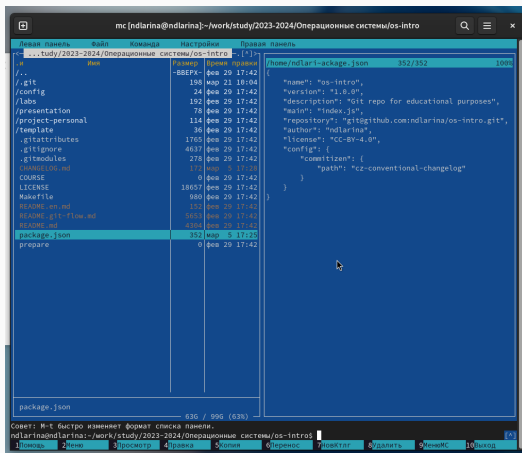


Рис. 7: Быстрый просмотр

# Работа с Midnight Commander

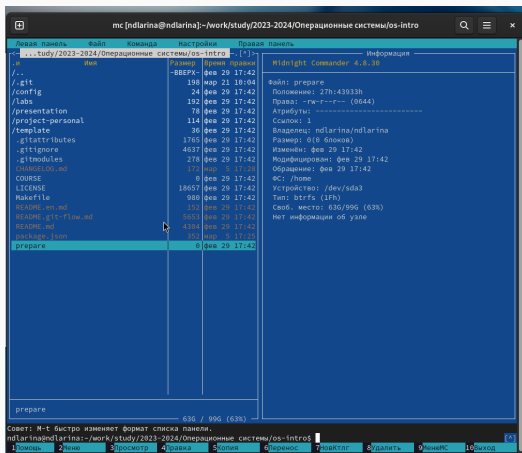


Рис. 8: Информация

# Работа с Midnight Commander

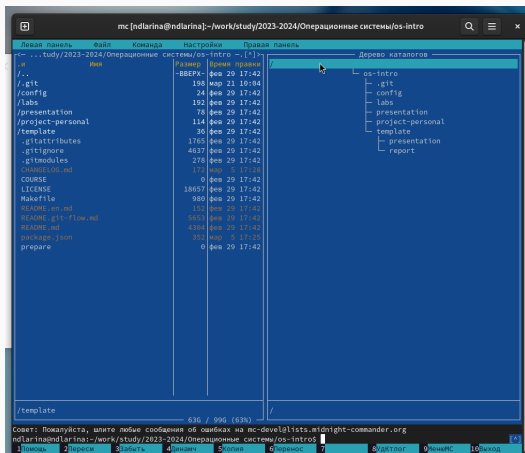
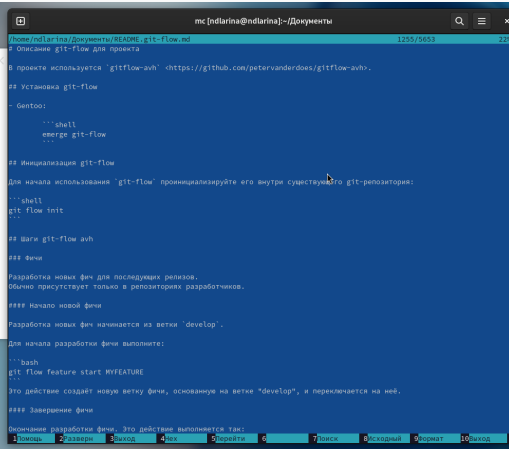


Рис. 9: Дерево каталогов

# Работа с Midnight Commander



The screenshot shows the Midnight Commander (mc) interface. The title bar indicates the user is 'ndlarina@ndlarina' in the directory '~/.Документы'. The current file is '/home/ndlarina/Документы/README\_git-flow.md' with a size of 1255/5653 and a date of 22. The file content is displayed in a monospaced font on a dark blue background. The content describes the 'git-flow-avh' workflow, including installation instructions for Gentoo, initialization steps, and a list of features. At the bottom, there is a navigation bar with buttons: 1.Помощь, 2.Заверши, 3.Выход, 4.Нав, 5.Порейти, 6., 7.Поиск, 8.Восходный, 9.Формат, 10.Выход.

```
mc [ndlarina@ndlarina]:~/.Документы
/home/ndlarina/Документы/README_git-flow.md 1255/5653 22
# Описание git-flow для проекта

В проекте используется 'gitflow-avh' <https://github.com/petervanderdoes/gitflow-avh>.

## Установка git-flow

- Gentoo:

    ```shell
    emerge git-flow
    ```

## Инициализация git-flow

Для начала использования 'git-flow' проинициализируйте его внутри существующего git-репозитория:

    ```shell
    git flow init
    ```

## Шаги git-flow avh

### Фичи

Разработка новых фич для последующих релизов.
Обычно присутствует только в репозиториях разработчиков.

#### Начало новой фичи

Разработка новых фич начинается из ветки 'develop'.

Для начала разработки фичи выполните:

    ```bash
    git flow feature start MYFEATURE
    ```

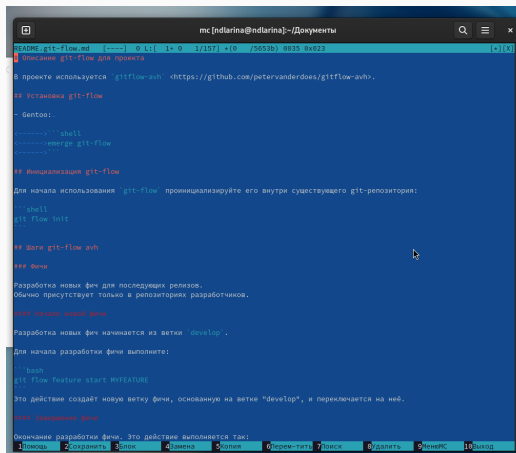
Это действие создаст новую ветку фичи, основанную на ветке "develop", и переключается на неё.

#### Завершение фичи

Окончание разработки фичи. Это действие выполняется так:
```

Рис. 10: Просмотр содержимого текстового файла

# Работа с Midnight Commander



**Рис. 11:** Отредактируем содержимое текстового файла без сохранения результатов



# Работа с Midnight Commander

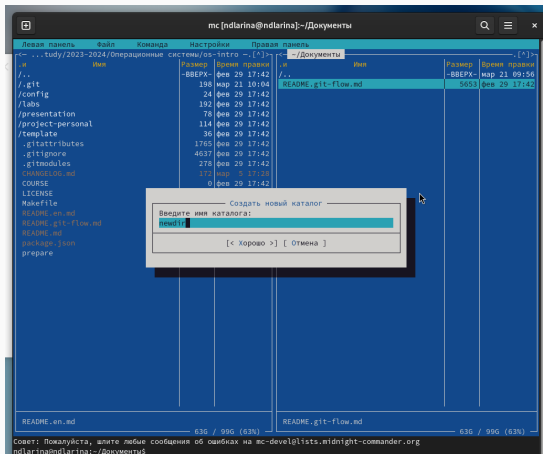


Рис. 12: Создание каталога

# Работа с Midnight Commander

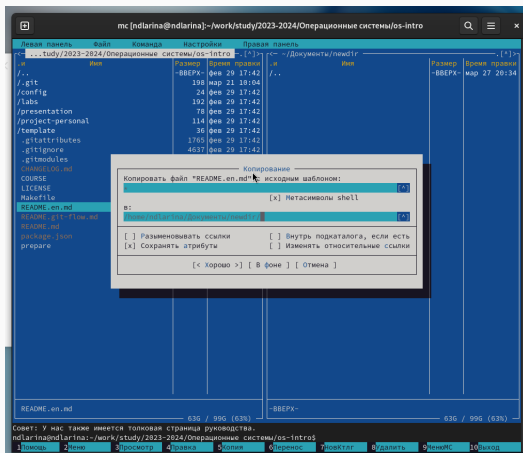


Рис. 13: Копирование в файлов в созданный каталог

# Работа с Midnight Commander

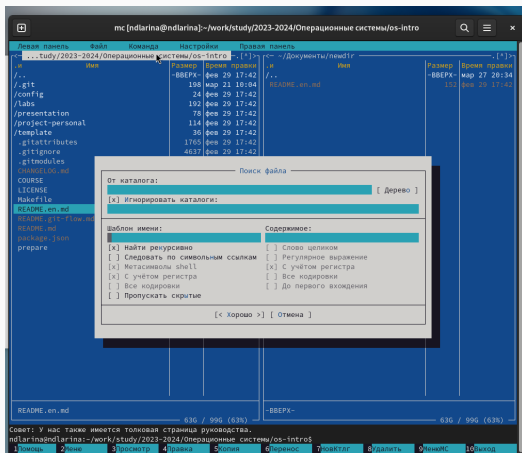


Рис. 14: Поиск файлов

# Работа с Midnight Commander

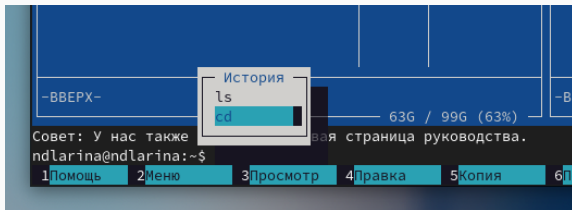
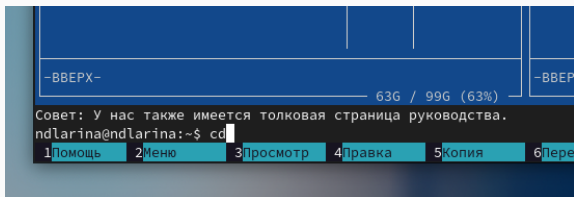


Рис. 15: История команд



**Рис. 16:** Переход в домашний каталог

# Работа с Midnight Commander



```
mc.ext.ini [-----] 0 L: [ 1+ 0 1/1136] * (0 /26909b) 0035 0x023 [.*][X]
# Midnight Commander 4.8 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 (F8-c-e). If you edit this file in any other way
# while mc is running, you will need to press F8-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 as before as after sections that point to it.
#
# Sections are processed from top to bottom, thus the order is important.
# If there are more than one sections with the same name in this file, the first
# section will be used.
#
# [Default] should be a catch-all action and come last.
#
```

Рис. 17: Просмотр файла расширений

A screenshot of the Midnight Commander file menu. The window title is "menu" with a status bar at the bottom showing "1: Помощь 2: Сохранить 3: Поиск 4: Имена 5: Скопировать 6: Перенести 7: Повторить 8: Удалить 9: Найти 10: Выход". The menu content includes options like "shell\_patterns=0", "#####", "F The % character", "f The current file", "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)", "v view Runs the commands and pipes standard output to the view command", "If vview 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", "f don't quote expanded macro", "#####", "t t Do something on the current file", "CMD=(Enter command)", "SCMD %f", "t t Do something on the tagged files", "CMD=(Enter command)", "for i in %t ; do".

```
menu [-----] 1 t: 1*16 17/378 *(732 /11821b) 0010 0x00A [x]
shell_patterns=0

#####
# % The % character
# f The current file (if non-local vfs, file will be copied locally and
# % 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)
# v view Runs the commands and pipes standard output to the view command
# If vview 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
# %f don't quote expanded macro
#####

+ t t
| Do something on the current file
| CMD=(Enter command)
| SCMD %f

+ t t
| Do something on the tagged files
| CMD=(Enter command)
| for i in %t ; do
```

Рис. 18: Просмотр файла меню

# Работа с Midnight Commander



Рис. 19: Конфигурация



# Работа с Midnight Commander



Рис. 20: Внешний вид

# Работа с Midnight Commander



Рис. 21: Настройки панелей



Рис. 22: Подтверждение



**Рис. 23:** Оформление



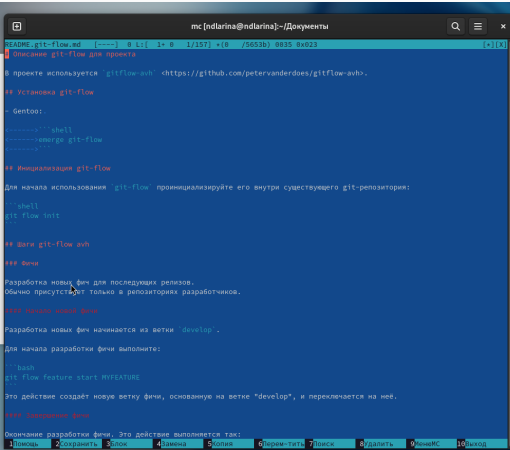
**Рис. 24:** Кодировка символов

# Работа с Midnight Commander



Рис. 25: Распознавание клавиш

# Работа с редактором Midnight Commander



The screenshot shows the Midnight Commander (mc) interface. The title bar indicates the user is 'mc [ndlarina@ndlarina] ~/Документы'. The file being edited is 'README\_git-flow.md'. The text content of the file is as follows:

```
Описание git-flow для проекта

В проекте используется 'gitflow-avh' <https://github.com/petervanderdoes/gitflow-avh>.

## Установка git-flow

- Gentoo:

-----> shell
-----> emerge git-flow
----->

## Инициализация git-flow

Для начала использования 'git-flow' проинициализируйте его внутри существующего git-репозитория:

    shell
    git flow init

## Заги git-flow avh

### Фичи

Разработка новых фич для последующих релизов.
Обычно присутствует только в репозиториях разработчиков.

### Начало новой фичи

Разработка новых фич начинается из ветки 'develop'.

Для начала разработки фичи выполните:

    bash
    git flow feature start MYFEATURE

Это действие создаст новую ветку фичи, основанную на ветке 'develop', и переключается на неё.

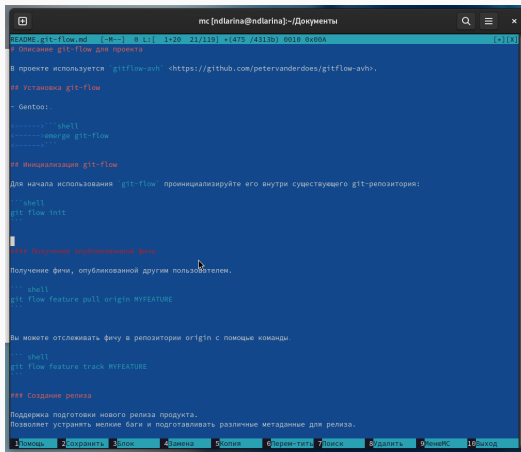
### Завершение фичи

Описание разработки фичи. Это действие выполняется так:
```

At the bottom of the window, there is a toolbar with icons for various actions: 'Помощь' (Help), 'Сохранить' (Save), 'Блок' (Block), 'Замена' (Replace), 'Копия' (Copy), 'Перем-тить' (Move), 'Поиск' (Search), 'Удалить' (Delete), 'Формат' (Format), and 'Выход' (Exit).

Рис. 26: Файл с текстом

# Работа с редактором Midnight Commander



```
mc [ndlarina@ndlarina] ~/Документы
README-git-flow.md [-M--] 9 L: 1+20 22/119 [+475 /4312b] 0010 0x00A [+] [X]
# Описание git-flow для проекта

В проекте используется 'gitflow-avh' <https://github.com/petervanderdoes/gitflow-avh>.

## Установка git-flow

- Gentoo:



```

-----''' shell
-----''' emerge git-flow
-----'''

```



## Инициализация git-flow

Для начала использования 'git-flow' проинициализируйте его внутри существующего git-репозитория:



```

''' shell
git flow init

```



### Получение опубликованной фичи

Получение фичи, опубликованной другим пользователем.



```

''' shell
git flow feature pull origin MYFEATURE

```



Вы можете отслеживать фичу в репозитории origin с помощью команды.



```

''' shell
git flow feature track MYFEATURE

```



## Создание релиза

Поддержка подготовки нового релиза продукта.
Позволяет устранить мелкие баги и подготавливать различные метаданные для релиза.

[Помощь] [Сохранить] [Слов] [Замена] [Копия] [Вперед-титл] [Поиск] [Удалить] [Меню] [Выход]
```

Рис. 27: Файл с текстом



# Работа с редактором Midnight Commander

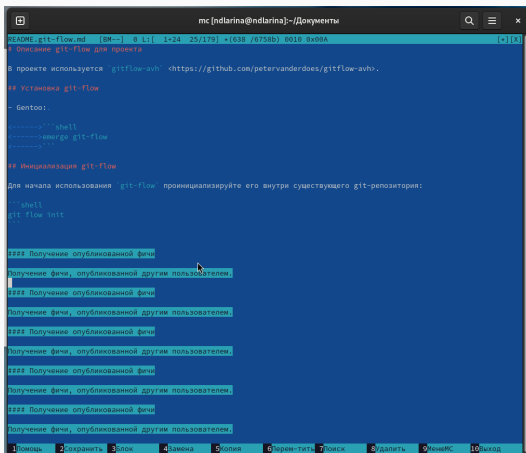


Рис. 28: Копирование фрагмента

# Работа с редактором Midnight Commander

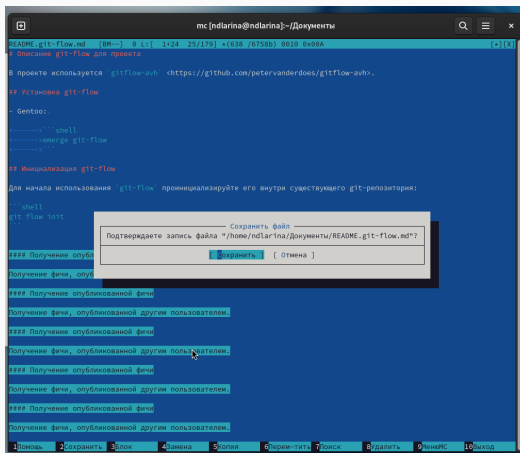
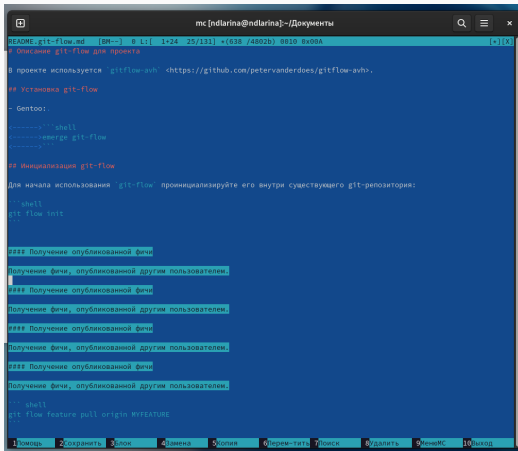


Рис. 29: Сохранение

# Работа с редактором Midnight Commander



```
mc [ndlarina@ndlarina:~/Документы
README.git-flow.md [BM--] 8 Li: 1+24 25/131] +{638 /4802b) 8010 8x08A
# Описание git-flow для проекта

В проекте используется 'git-flow-avh' <https://github.com/petervanderdoes/gitflow-avh>.

## Установка git-flow

- Gentoo:
  ~~~~~''' shell
  ~~~~~merge git-flow
  ~~~~~'''

## Инициализация git-flow

Для начала использования 'git-flow' проинициализируйте его внутри существующего git-репозитория:

''' shell
git flow init

#### Получение опубликованной фичи

Получение фичи, опубликованной другим пользователем.
#### Получение опубликованной фичи
Получение фичи, опубликованной другим пользователем.
#### Получение опубликованной фичи
Получение фичи, опубликованной другим пользователем.
#### Получение опубликованной фичи
Получение фичи, опубликованной другим пользователем.
#### Получение опубликованной фичи
Получение фичи, опубликованной другим пользователем.

''' shell
git flow feature pull origin MYFEATURE

1 Помощь 2 Сохранить 3 Вывод 4 Замена 5 Копия 6 Переименовать 7 Поиск 8 Удалить 9 Меню 10 Выход
```

Рис. 30: Отмена

# Работа с редактором Midnight Commander

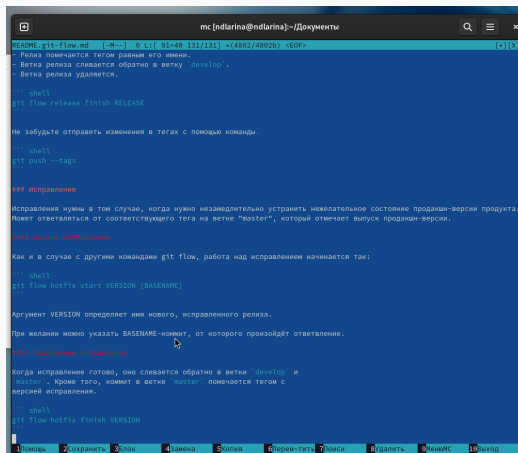
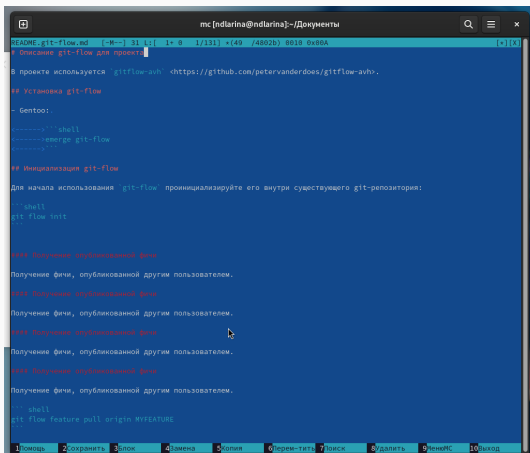


Рис. 31: Переход в конец файла

# Работа с редактором Midnight Commander



```
mc [ndlarina@ndlarina] ~/Документы
README.git-flow.md [-M--] 21 1:1 1* 0 1/131] x(49 /4802b) 8018 0x80A [-] [X]
# Описание git-flow для проекта

В проекте используется 'gitflow-avh' <https://github.com/petervanderdoes/gitflow-avh>.

## Установка git-flow

- Gentoo:



```

-----''' shell
-----''' emerge git-flow
-----'''

```



## Инициализация git-flow

Для начала использования 'git-flow' проинициализируйте его внутри существующего git-репозитория:



```

''' shell
git flow init

```



#### Получение опубликованной фичи

Получение фичи, опубликованной другим пользователем.

#### Получение опубликованной фичи

Получение фичи, опубликованной другим пользователем.

#### Получение опубликованной фичи

Получение фичи, опубликованной другим пользователем.

#### Получение опубликованной фичи

Получение фичи, опубликованной другим пользователем.

#### Получение опубликованной фичи

Получение фичи, опубликованной другим пользователем.



```

''' shell
git flow feature pull origin MYFEATURE

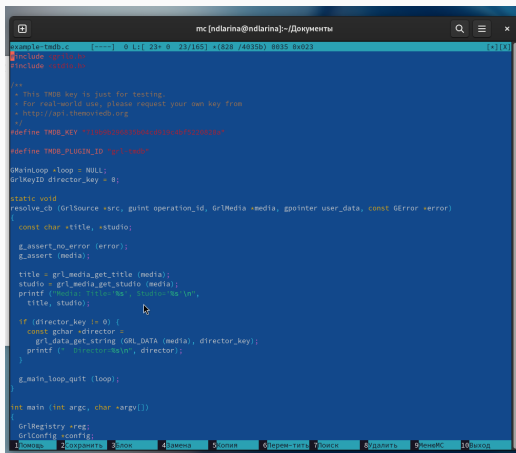
```



1 Помощь 2 Сохранить 3 Блок 4 Замена 5 Копия 6 Переименовать 7 Поиск 8 Удалить 9 Меню 10 Выход
```

Рис. 32: Переход в начало файла

# Работа с редактором Midnight Commander



```
example-tmdb.c  [-----]  0 L: [ 23* 0  23/165]  *(828 /4035b) 0035 0x023  [x] [X]
#include <gstreamer/gstreamer.h>
#include <stdio.h>

/**
 * This TMDB key is just for testing.
 * For real-world use, please request your own key from
 * http://api.themoviedb.org
 */
#define TMDB_KEY "1234567890123456789012345678901234567890"

#define TMDB_PLUGIN_ID "gri-tmdb"

GMainLoop *loop = NULL;
GrlKeyID director_key = 0;

static void
resolve_cb (GrlSource *src, guint operation_id, GrlMedia *media, gpointer user_data, const GError *error)
{
    const char *title, *studio;

    g_assert_no_error (error);
    g_assert (media);

    title = grl_media_get_title (media);
    studio = grl_media_get_studio (media);
    printf ("media: %s, title: %s, studio: %s\n",
           title, studio);

    if (director_key != 0) {
        const gchar *director =
            grl_data_get_string (GRL_DATA (media), director_key);
        printf ("director: %s\n", director);
    }

    g_main_loop_quit (loop);
}

int main (int argc, char *argv[])
{
    GrlRegistry *reg;
    GrlConfig *config;

    1.Помощь 2.Сохранить 3.Файл 4.Замена 5.Копия 6.Перенести 7.Поиск 8.Удалить 9.Выход
```

Рис. 33: Файл с программой

# Работа с редактором Midnight Commander

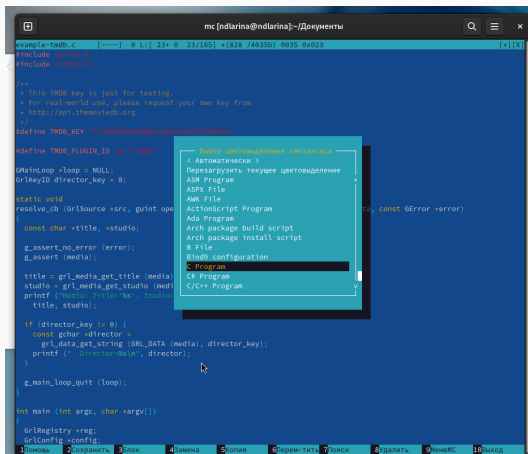


Рис. 34: Цветовыделение синтаксиса

## **Выводы по проделанной работе**

---



В данной работе мы ознакомились с инструментами поиска файлов и фильтрации текстовых данных. А также приобрели практические навыки по управлению процессами.