

# Командная оболочка Midnight Commander

---

Раджаб Раджабов<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

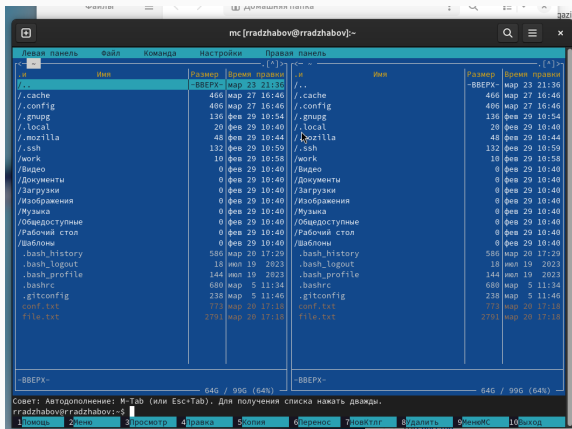


Рис. 1: Запуск mc

# Работа с Midnight Commander

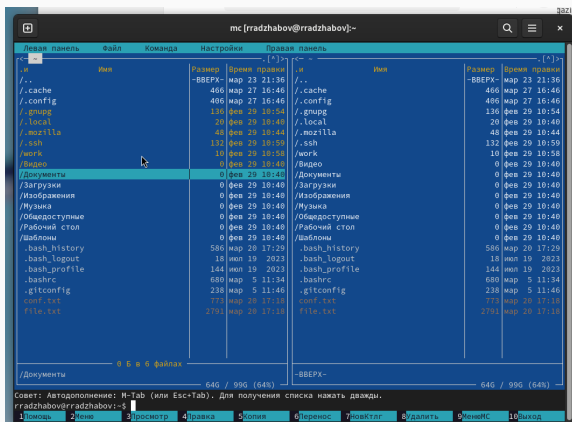


Рис. 2: Выделение

# Работа с Midnight Commander

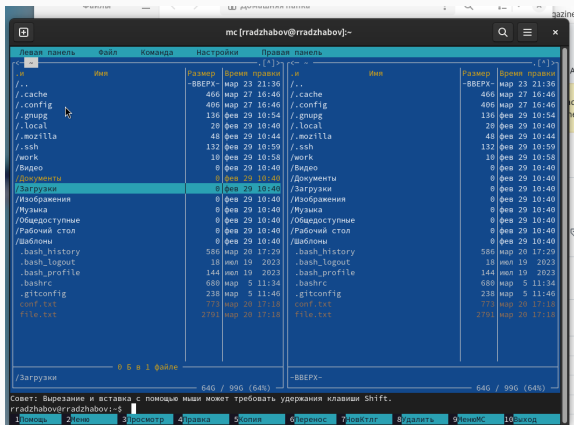


Рис. 3: Отмена



# Работа с Midnight Commander

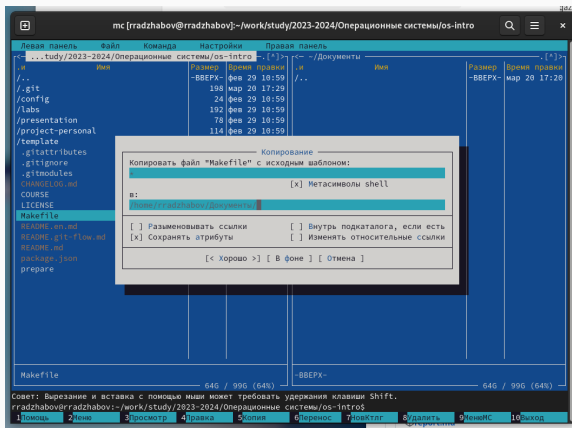


Рис. 4: Копирование

# Работа с Midnight Commander

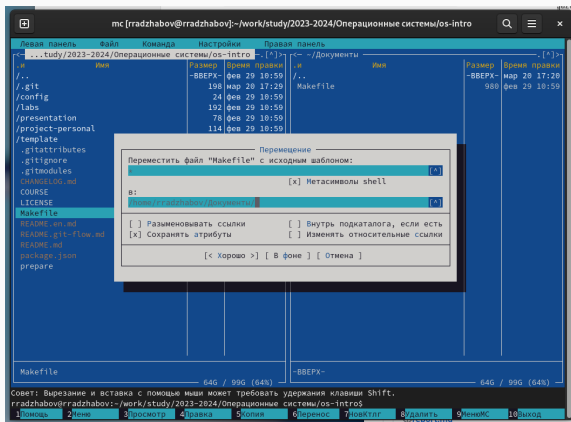


Рис. 5: Перемещение

# Работа с Midnight Commander

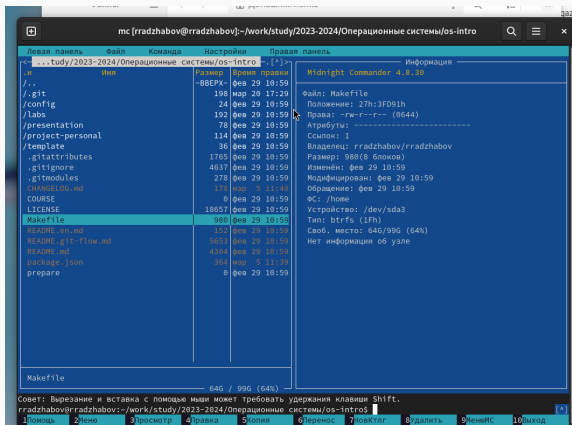


Рис. 6: Информация

# Работа с Midnight Commander

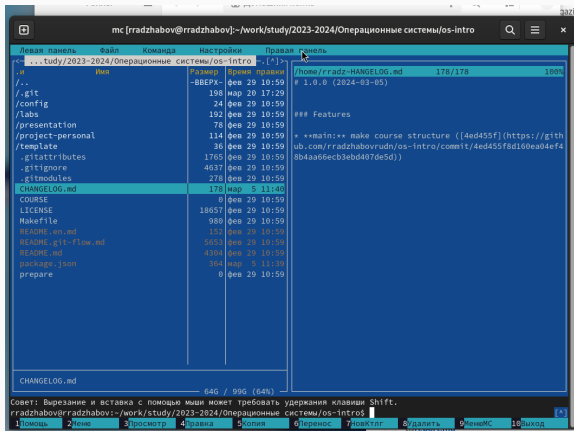


Рис. 7: Быстрый просмотр

# Работа с Midnight Commander

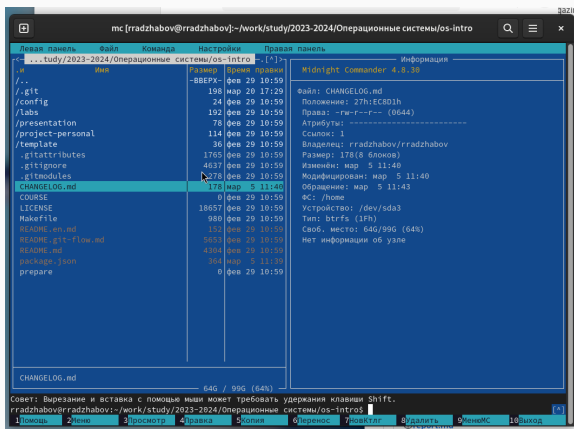


Рис. 8: Информация

# Работа с Midnight Commander

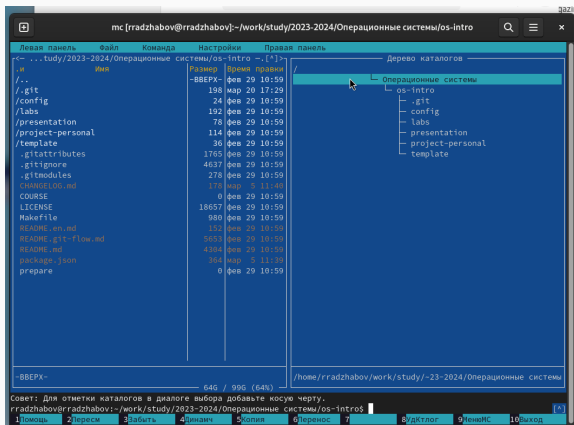
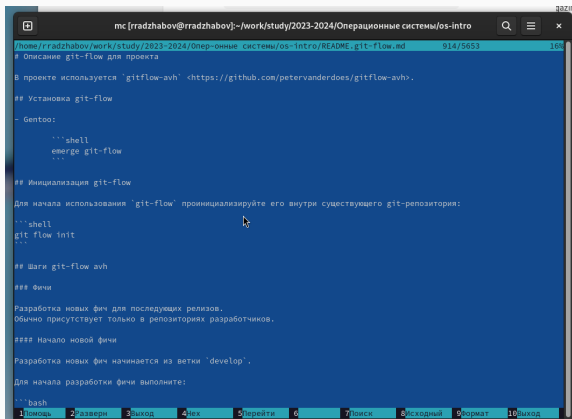


Рис. 9: Дерево каталогов

# Работа с Midnight Commander



The screenshot shows the Midnight Commander (MC) interface. The title bar indicates the current directory is `mc [rradzhahov@rradzhahov]:~/work/study/2023-2024/Операционные системы/os-intro`. The main window displays the contents of the file `/home/rradzhahov/work/study/2023-2024/Операционные системы/os-intro/README.git-flow.md`. The file content 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
    ```

## Шаги git-flow avh

### Фичи

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

#### Начало новой фичи

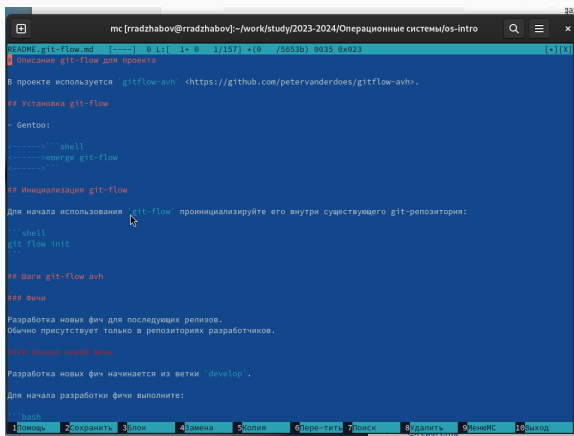
Разработка новых фич начинается из ветки 'develop'.
Для начала разработки фичи выполните:

    ```bash
```

At the bottom of the window, there is a navigation bar with buttons: 1.Помощь, 2.Разверн, 3.Выход, 4.Лек, 5.Перейти, 6., 7.Лист, 8.Исходный, 9.Формат, 10.Выход.

Рис. 10: Просмотр содержимого текстового файла

# Работа с Midnight Commander



The screenshot shows the Midnight Commander (MC) interface. The title bar indicates the user is 'mc [rradzhahov@rradzhahov]~ /work/study/2023-2024/Операционные системы/os-intro'. The main window displays the contents of a file named 'README.git-flow.md'. The text in the file is as follows:

```
README.git-flow.md  [----]  0 L:  1+ 0  1/157)  *(0  /5653b)  0035  0x023  [*)(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

## Запуск 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
```

At the bottom of the window, there is a toolbar with icons and labels for various actions: 1. Помощь (Help), 2. Сохранить (Save), 3. Залок (Lock), 4. Замена (Replace), 5. Копия (Copy), 6. Пере-тить (Paste), 7. Поиск (Search), 8. Удалить (Delete), 9. Переименовать (Rename), 10. Выход (Exit).

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



# Работа с Midnight Commander

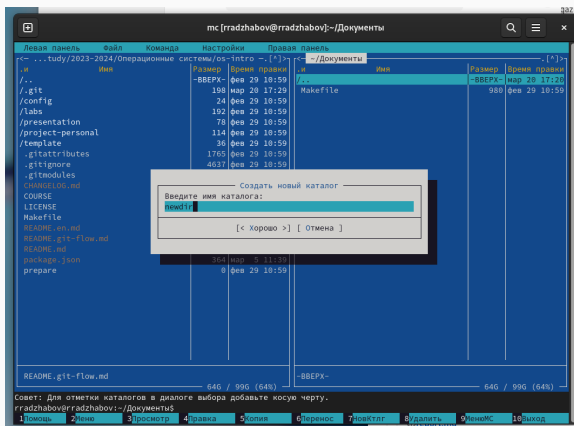


Рис. 12: Создание каталога

# Работа с Midnight Commander

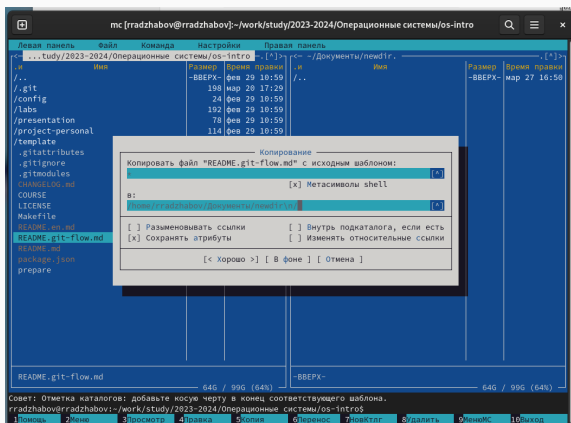


Рис. 13: Копирование в файлов в созданный каталог

# Работа с Midnight Commander

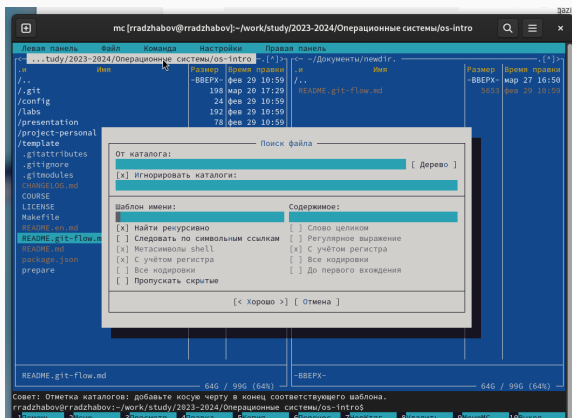


Рис. 14: Поиск файлов

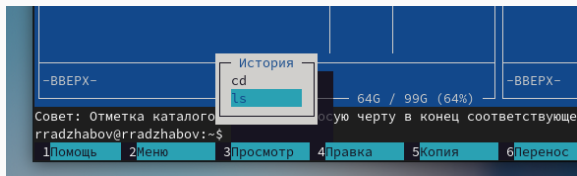


Рис. 15: История команд

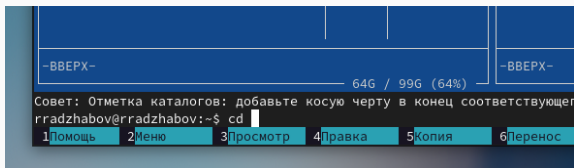


Рис. 16: Переход в домашний каталог

# 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: Скопия 6: Переименовать 7: Поиск 8: Удалить 9: Ненавс 10: Выход". The menu content includes options like "shell\_patterns=0", "#####", "% 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)", "vview 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", and a status bar at the bottom with 10 numbered buttons: 1: Помощь, 2: Сохранить, 3: Поиск, 4: Имена, 5: Скопия, 6: Переименовать, 7: Поиск, 8: Удалить, 9: Ненавс, 10: Выход.

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



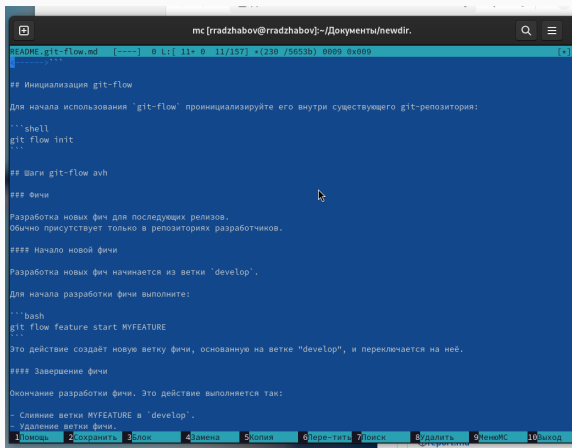
**Рис. 24:** Кодировка символов

# Работа с Midnight Commander



Рис. 25: Распознавание клавиш

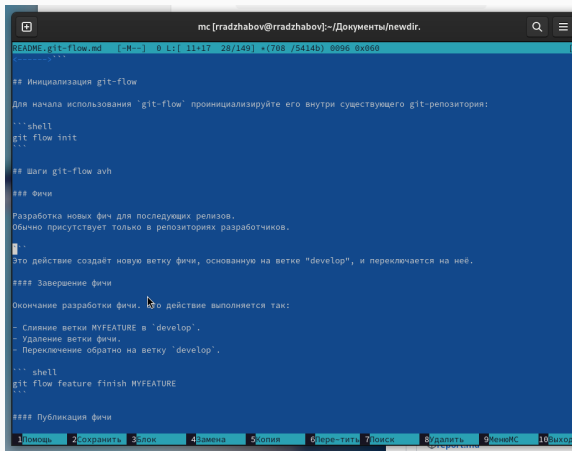
# Работа с редактором Midnight Commander



```
mc [rradzhabov@rradzhabov]:~/Документы/newdir.  
README.git-flow.md [----] 0 L: 11+ 0 11/157 + (230 /5653b) 0009 0x009 [•]  
#####  
## Инициализация git-flow  
Для начала использования 'git-flow' проинициализируйте его внутри существующего git-репозитория:  
```shell  
git flow init  
```  
## Шаг 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:  
- Слияние ветки MYFEATURE в 'develop'.  
- Удаление ветки фичи.  
1Помощь 2Сохранить 3Блок 4Замена 5Копия 6Пере-ить 7Поиск 8Удалить 9Меню 10Выход
```

Рис. 26: Файл с текстом

# Работа с редактором Midnight Commander



```
mc [rradzhbov@rradzhbov]:~/Документы/newdir.
README.git-flow.md [-M--] 0 L: [11+17 28/149] +(708 /5414b) 8096 0x060

## 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.

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

#### Завершение фичи

Окончание разработки фичи. Это действие выполняется так:

- Слияние ветки MYFEATURE в 'develop'.
- Удаление ветки фичи.
- Переключение обратно на ветку 'develop'.

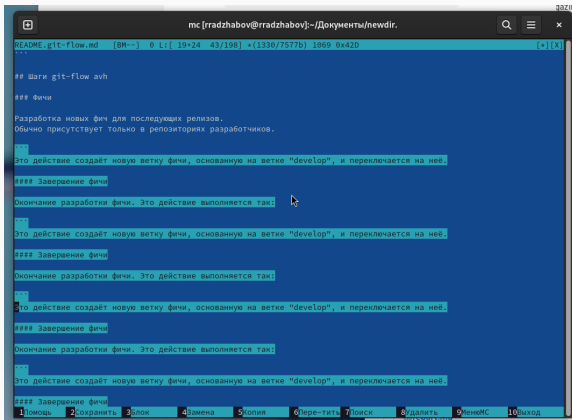
''' shell
git flow feature finish MYFEATURE

#### Публикация фичи
```

Рис. 27: Файл с текстом



## Работа с редактором Midnight Commander



### Рис. 28: Копирование фрагмента

# Работа с редактором Midnight Commander

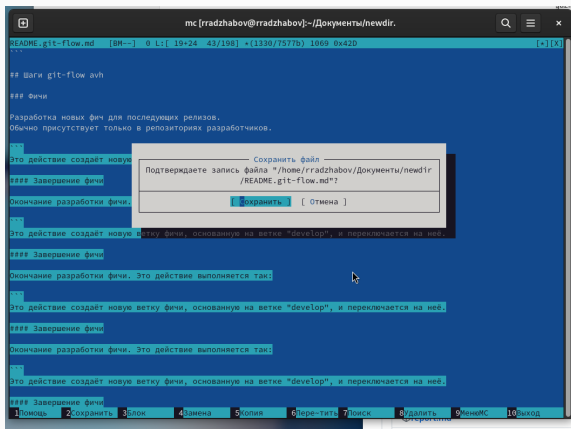
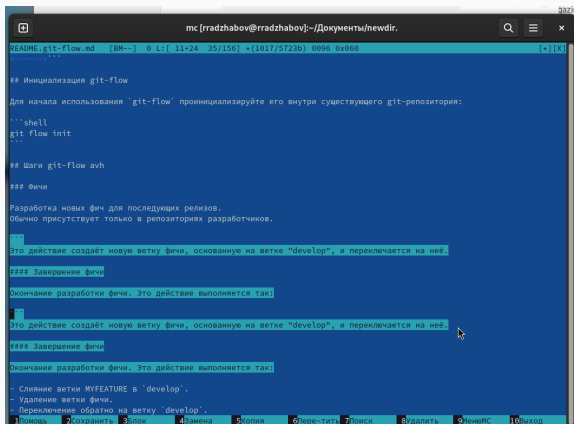


Рис. 29: Сохранение

# Работа с редактором Midnight Commander



The screenshot shows the Midnight Commander (mc) interface. The title bar indicates the current directory is ~/Документы/newdir. The main window displays the contents of a file named README.git-flow.md. The text within the file describes the git-flow workflow, including initialization, branching rules, and commands for creating and finishing branches. The interface has a dark blue background with light blue text. At the bottom, there is a menu bar with options like Помощь, Сохранить, Эпик, Заменить, Скопировать, Переименовать, Поиск, Удалить, Переименовать, and Выход.

```
mc [rradzhbov@rradzhbov]-:~/Документы/newdir.
README.git-flow.md [BM--] 0 L: 11:24 35/156 *(1017/5723b) 0096 0x060 [+][X]

## 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.

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

### Завершение фичи

Окончание разработки фичи. Это действие выполняется так:

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

### Завершение фичи

Окончание разработки фичи. Это действие выполняется так:

- Слияние ветки MYFEATURE в 'develop'.
- Удаление ветки фичи.
- Переключение обратно на ветку 'develop'.

1 Помощь 2 Сохранить 3 Эпик 4 Заменить 5 Скопировать 6 Переименовать 7 Поиск 8 Удалить 9 Переименовать 10 Выход
```

Рис. 30: Отмена

# Работа с редактором Midnight Commander

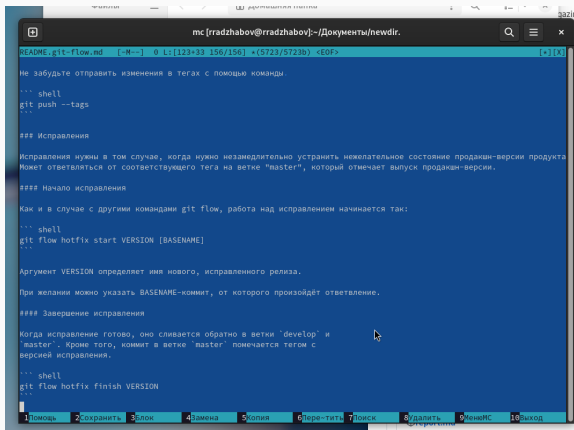


Рис. 31: Переход в конец файла

# Работа с редактором Midnight Commander

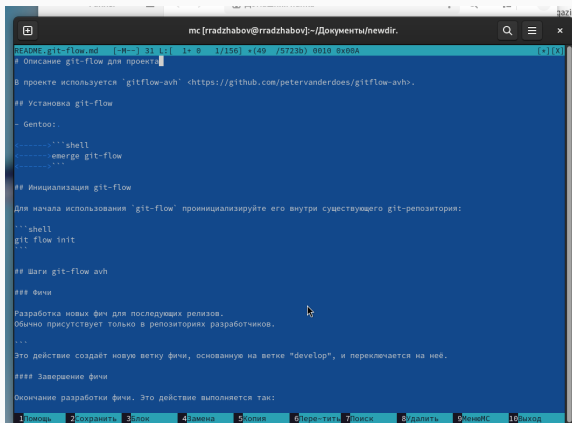
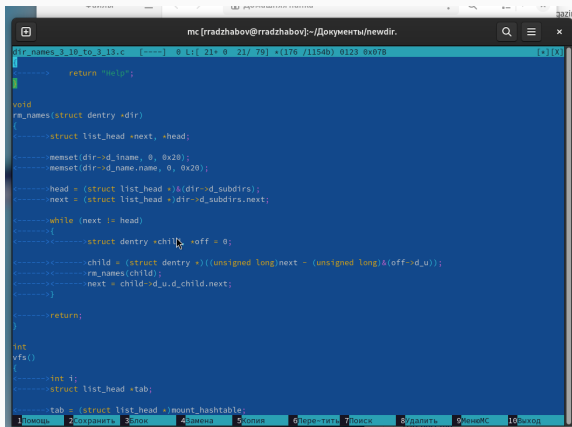


Рис. 32: Переход в начало файла

# Работа с редактором Midnight Commander



```
mc [rradzhbov@rradzhbov]~/Документы/newdir.  
dir_names_3_10_to_3_13.c  [----]  0 1: 21* 0 21/ 79) *(176 /1154b) 0123 8x078  [=] [X]  
-----> return "help";  
  
void  
rm_names(struct dentry *dir)  
{  
-----> struct list_head *next, *head;  
  
-----> memset(dir->d_name, 0, 0x20);  
-----> memset(dir->d_name.name, 0, 0x20);  
  
-----> head = (struct list_head *)&(dir->d_subdirs);  
-----> next = (struct list_head *)&(dir->d_subdirs.next);  
  
-----> while (next != head)  
-----> {  
----->-----> struct dentry *child; *off = 0;  
  
----->-----> child = (struct dentry *)((unsigned long)next - (unsigned long)&(off->d_u));  
----->-----> rm_names(child);  
----->-----> next = child->d_u.d_child.next;  
----->-----> }  
  
-----> return;  
}  
  
int  
vfs()  
{  
-----> int i;  
-----> struct list_head *tab;  
  
-----> tab = (struct list_head *)&mount_hashtable;
```

Рис. 33: Файл с программой

# Работа с редактором Midnight Commander

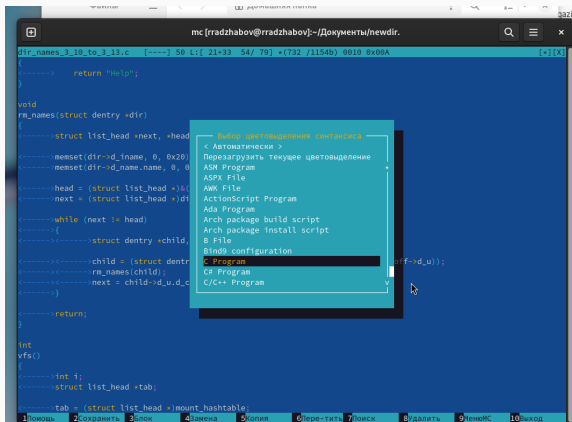


Рис. 34: Цветовыделение синтаксиса

## **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.