

Командная оболочка Midnight Commander

Дмитрий Джало ¹

27 марта, 2024, Москва, Россия

¹Российский 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

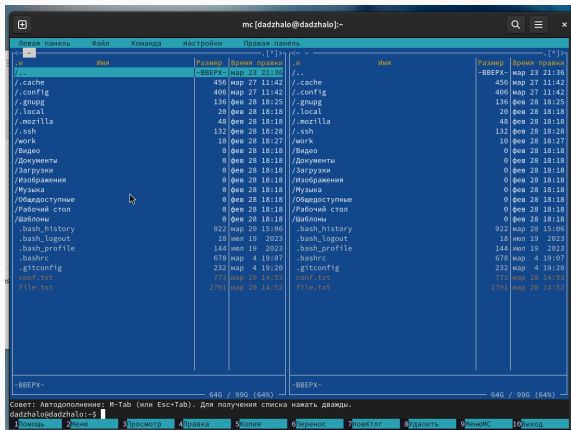


Рис. 1: Запуск mc

Работа с Midnight Commander

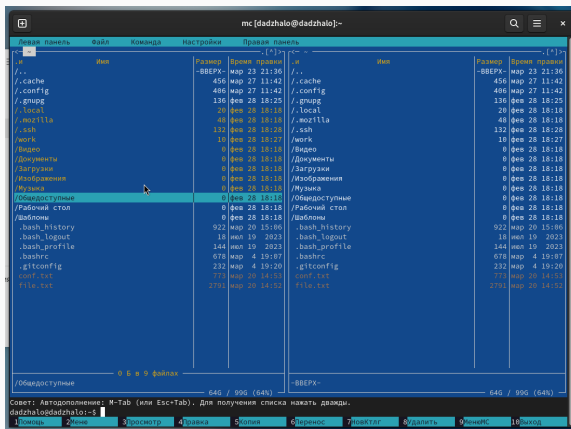


Рис. 2: Выделение

Работа с Midnight Commander

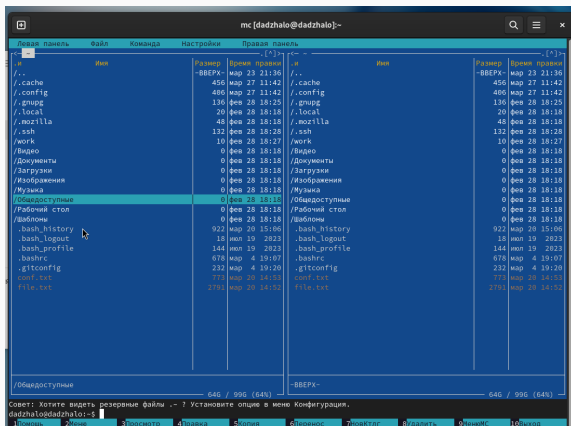


Рис. 3: Отмена

Работа с Midnight Commander

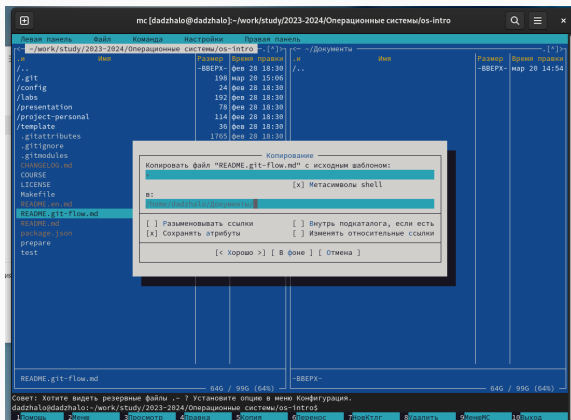


Рис. 4: Копирование

Работа с Midnight Commander

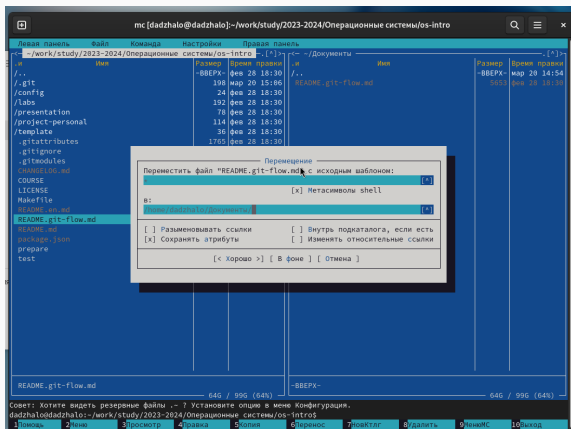


Рис. 5: Перемещение

Работа с Midnight Commander

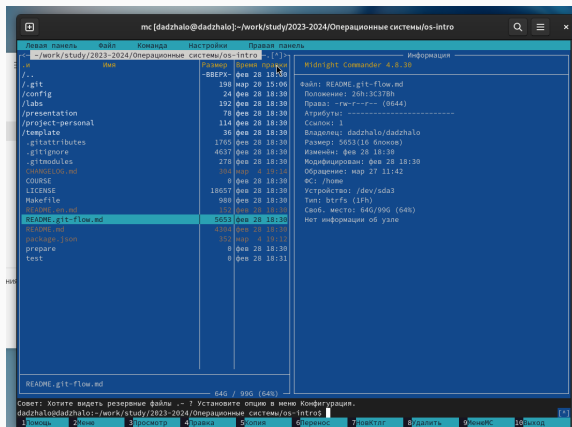


Рис. 6: Информация

Работа с Midnight Commander

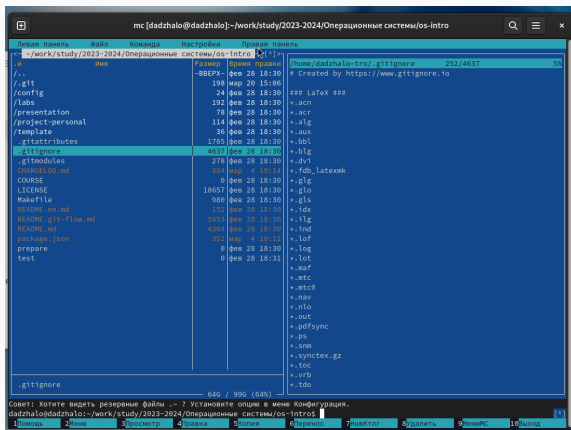


Рис. 7: Быстрый просмотр

Работа с Midnight Commander

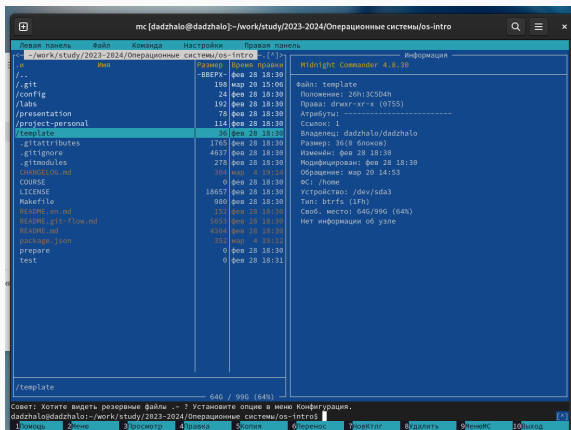


Рис. 8: Информация

Работа с Midnight Commander

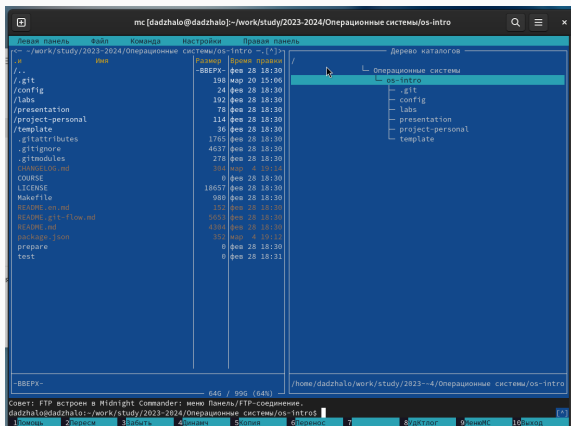
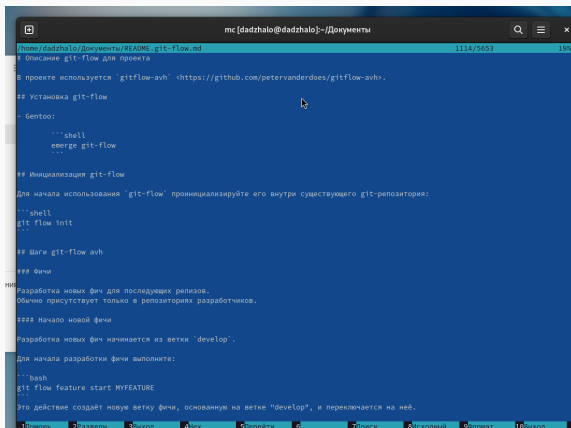


Рис. 9: Дерево каталогов

Работа с Midnight Commander



The screenshot shows the Midnight Commander interface with a terminal window open. The terminal displays the contents of the file `/home/dadzhalo/Документы/README_git-flow.md`. The text in the terminal is as follows:

```
mc [dadzhalo@dadzhalo]~/Документы
/home/dadzhalo/Документы/README_git-flow.md 1114/5653 19%
# 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:

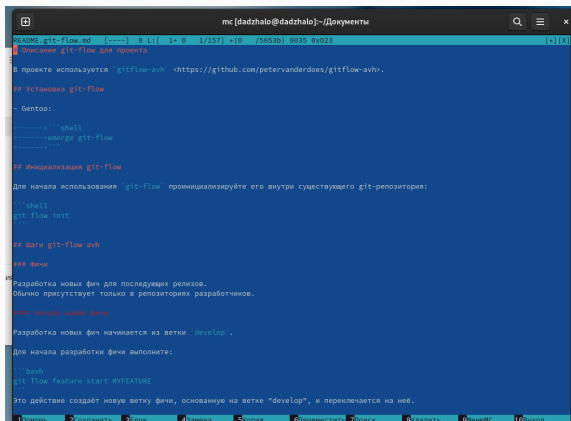
'''bash
git flow feature start MYFEATURE
...

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

At the bottom of the terminal window, there is a navigation bar with buttons: `Помощь`, `Разверн`, `Выход`, `Фичи`, `Ветвей`, `с`, `Поиск`, `Всходный`, `Формат`, `Выход`.

Рис. 10: Просмотр содержимого текстового файла

Работа с Midnight Commander



The screenshot shows the Midnight Commander (mc) interface. The title bar indicates the current directory is ~/Документы. The main window displays the contents of the file README_git-flow.md. The text within the file describes the git-flow workflow, including installation instructions for Gentoo and the initialization of the workflow. At the bottom of the window, a toolbar contains 10 numbered icons for file operations: 1. Открыть (Open), 2. Сохранить (Save), 3. Вставить (Paste), 4. Заменить (Replace), 5. Копировать (Copy), 6. Переместить (Move), 7. Поиск (Search), 8. Удалять (Delete), 9. Найти MC (Find MC), and 10. Выход (Exit).

```
mc [dadzhalo@dadzhalo] ~/Документы
README_git-flow.md [-----] 8 L: 3+ 0 1/157] + (0 /5653b) 0035 8x023 [~] [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

## Заги git-flow avh

## Фичи

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

git flow start new-feature

Разработка новых фич начинается из ветки 'develop'.

Для начала разработки фичи выполните:

" bash
git flow feature start MYFEATURE
"

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

Рис. 11: Отредактируем содержимое текстового файла без сохранения результатов

Работа с Midnight Commander

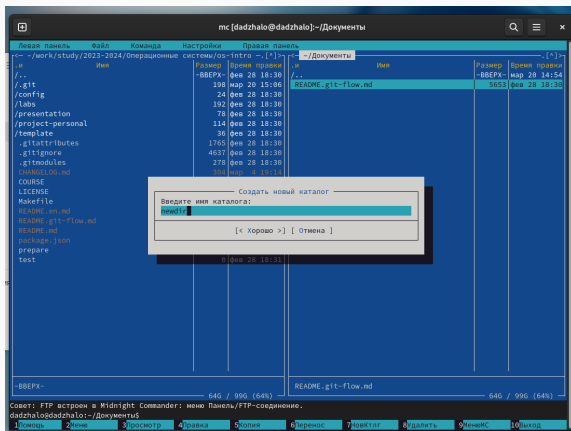


Рис. 12: Создание каталога

Работа с Midnight Commander

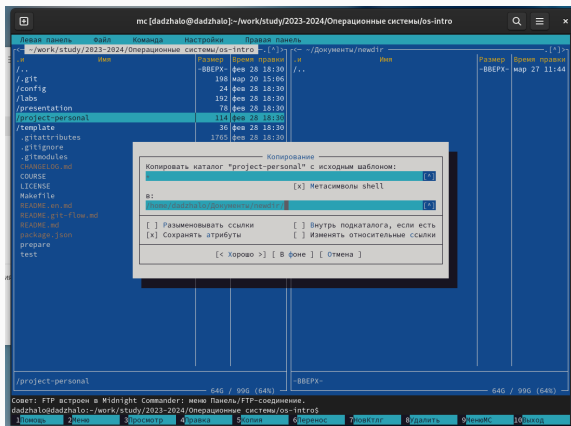


Рис. 13: Копирование в файлов в созданный каталог

Работа с Midnight Commander

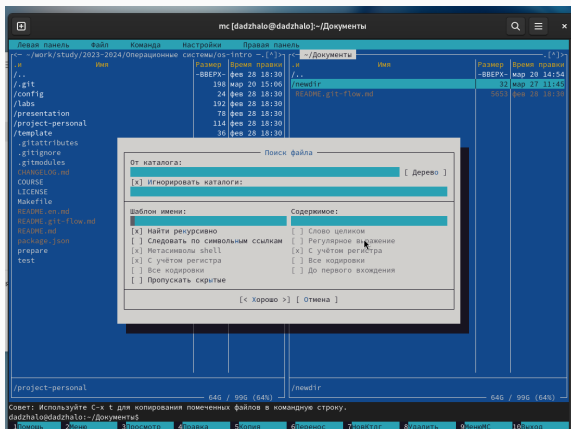


Рис. 14: Поиск файлов

Работа с Midnight Commander

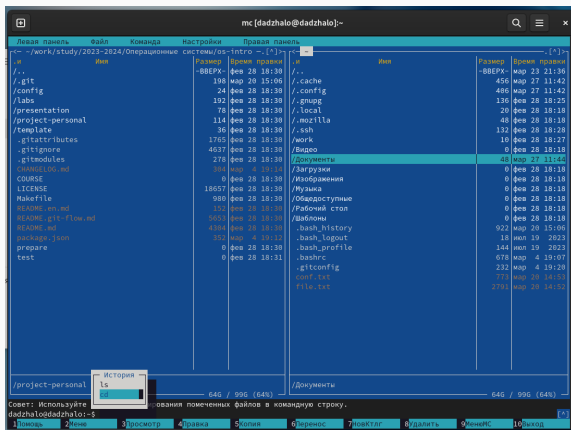


Рис. 15: История команд

Работа с Midnight Commander

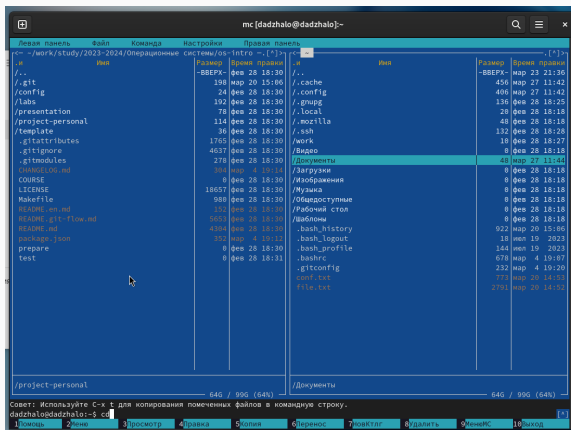


Рис. 16: Переход в домашний каталог

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

Работа с Midnight Commander

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: Перенести 7: Повтор 8: Удалить 9: НенавС 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 1 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: Перенести, 7: Повтор, 8: Удалить, 9: НенавС, 10: Выход.

```
menu [-----] 1 t: 1*16 17/370 *(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
0 Do something on the current file
CMD=: (Enter command)
SCMD: f

+ t t
0 Do something on the tagged files
CMD=: (Enter command)
for 1 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



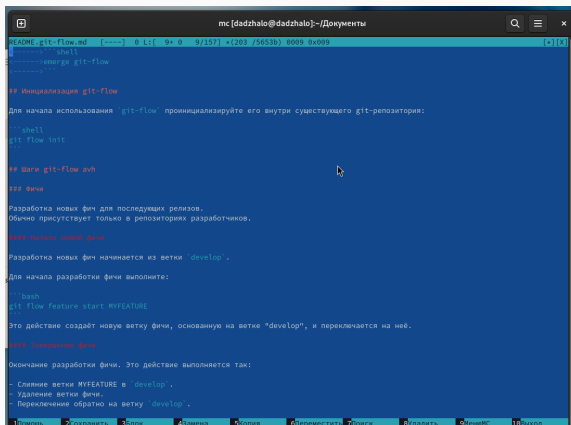
Рис. 24: Кодировка символов

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 current directory is ~/Документы and the file being edited is README.git-flow.md. The file content is as follows:

```
README.git-flow.md [-----] 0 Li[ 9+ 0 9/157] *(203 /5653b) 0009 0x009 [*](X)]
----- shell
----- merge git-flow
-----

## Инициализация git-flow

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

'''shell
git flow init
'''

## Work git-flow ash

## 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 создаёт новую ветку фичи, основанную на ветке "develop", и переключается на неё.

### Завершение фичи

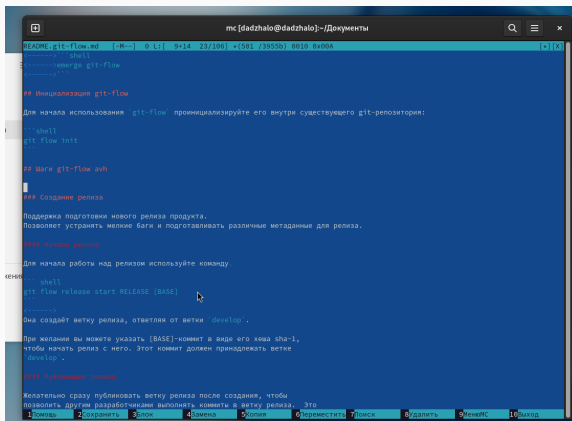
Окончание разработки фичи. Это действие выполняется так:

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

At the bottom of the window, there is a toolbar with icons for various actions: Помощь, Сохранить, Выход, Переименовать, Копия, Вставить, Поиск, Удалить, Меню, and Выход.

Рис. 26: Файл с текстом

Работа с редактором Midnight Commander



The screenshot shows the Midnight Commander (mc) file manager interface. The title bar indicates the user is 'dadzhalo' on a machine named 'dadzhalo', and the current directory is '/Документы'. The main window displays the contents of a file named 'README.git-flow.md'. The text in the file is as follows:

```
##### shell
##### merge git-flow
#####

## Инициализация git-flow

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

''' shell
git flow init

## Шаг git-flow avh

### Создание релиза

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

#### Начало релиза

Для начала работы над релизом используйте команду.

''' shell
git flow release start RELEASE [BASE]
'''

Она создаст ветку релиза, ответная от ветки 'develop'.

При желании вы можете указать [BASE]-коммит в виде его хеша sha-1,
чтобы начать релиз с него. Этот коммит должен принадлежать ветке
'develop'.

#### Публикация релиза

Желательно сразу публиковать ветку релиза после создания, чтобы
позволить другим разработчикам выполнять коммиты в ветку релиза. Это
```

At the bottom of the window, there is a toolbar with icons for various file operations: 1. Помочь (Help), 2. Сохранить (Save), 3. Блок (Block), 4. Замена (Replace), 5. Копия (Copy), 6. Переместить (Move), 7. Поиск (Find), 8. Удалить (Delete), 9. Меню (Menu), and 10. Выход (Exit).

Рис. 27: Файл с текстом

Работа с редактором Midnight Commander

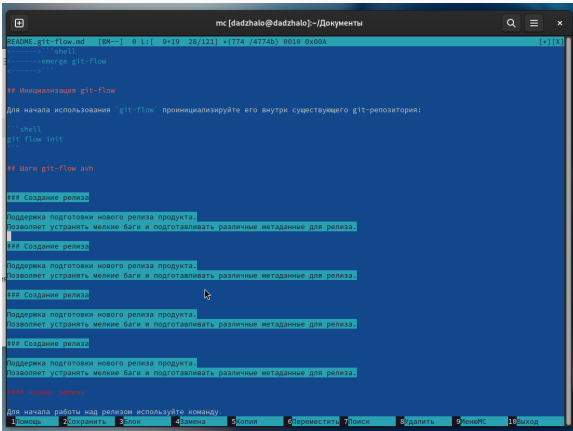


Рис. 28: Копирование фрагмента

Работа с редактором Midnight Commander

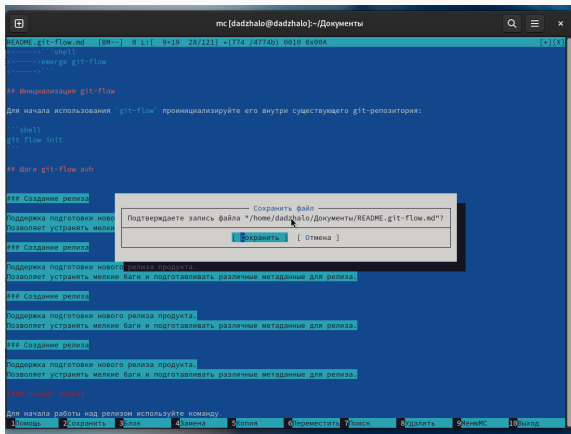
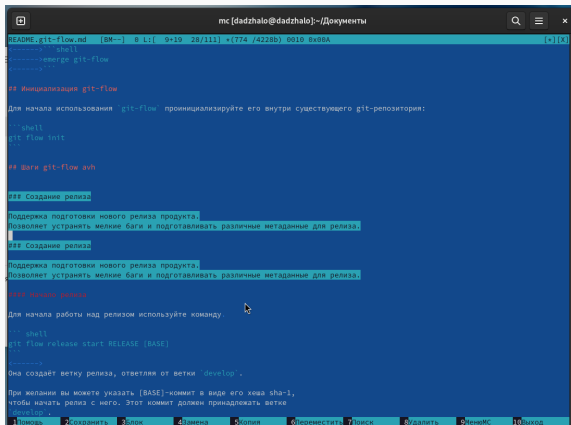


Рис. 29: Сохранение

Работа с редактором Midnight Commander



The screenshot shows the Midnight Commander (MC) interface. The title bar indicates the user is 'mc [dadzhalo@dadzhalo] -- /Документы'. The main window displays the contents of a file named 'README.git-flow.md'. The file content is as follows:

```
README.git-flow.md [8M--]  8 L:  9+19  28/111  + (774 /4228b) 6010 8x80A  [+][x]
-----> shell
-----> merge git-flow
-----> ...

## Инициализация git-flow

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

''' shell
git flow init
'''

## Заги git-flow avh

### Создание релиза

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

### Создание релиза

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

### Начало релиза

Для начала работы над релизом используйте команду:

''' shell
git flow release start RELEASE [BASE]
'''

Она создаёт ветку релиза, отвеля от ветки 'develop'.

При желании вы можете указать [BASE]-коммит в виде его sha-1,
чтобы начать релиз с него. Этот коммит должен принадлежать ветке
'develop'.
```

At the bottom of the window, there is a toolbar with icons for various actions: 'Назад' (Back), 'Сохранить' (Save), 'Открыть' (Open), 'Изменить' (Edit), 'Копировать' (Copy), 'Вставить' (Paste), 'Найти' (Find), 'Удалить' (Delete), 'Выход' (Exit), and 'Выход' (Exit).

Рис. 30: Отмена

Работа с редактором Midnight Commander

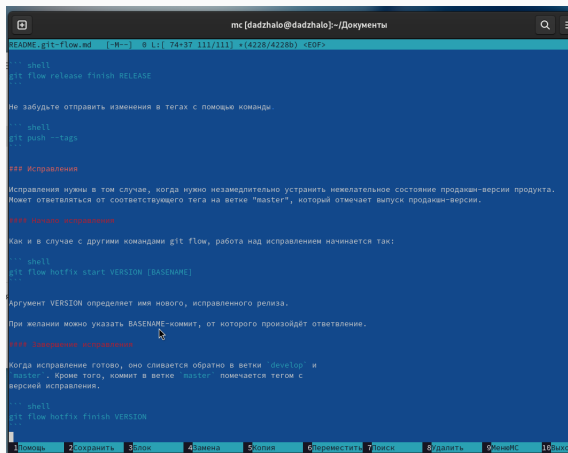


Рис. 31: Переход в конец файла

Работа с редактором Midnight Commander

```
mc [dadzhalo@dadzhalo] ~/Документы
README.git-flow.md [-M--] 31 L: 1+ B 1/111 + (49 /4228b) 0010 8x80A
+ Описание git-flow для проекта

В проекте используется 'git-flow-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
'''

## Шаг git-flow avh

### Создание релиза

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

### Создание релиза

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

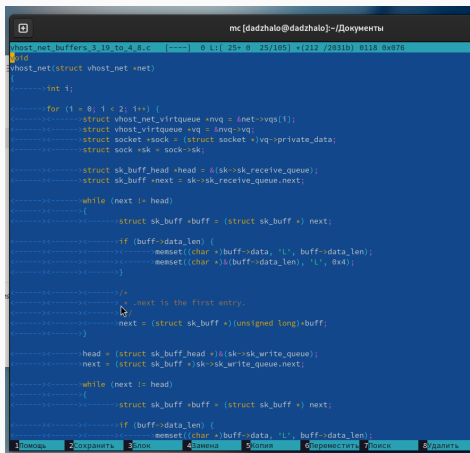
#### Начало релиза

Для начала работы над релизом используйте команду.

''' shell
1 Помощь 2 Сохранить 3 Блок 4 Замена 5 Скопировать 6 Переместить 7 Поиск 8 Удалить 9 Меню MC 10 Вы
```

Рис. 32: Переход в начало файла

Работа с редактором Midnight Commander



```
mc [dadzhalo@dadzhalo]:~/Документы
vhost_net_buffers,3,19,то,4,8,c  [----]  0 L: [ 25+ 0 25/105] *(212 /2831b) 8118 0x876
void
vhost_net(struct vhost_net *net)
{
    int i;

    for (i = 0; i < 2; i++) {
        struct vhost_net_virtqueue *nvq = &net->vqs[i];
        struct vhost_virtqueue *vq = &nvq->vq;
        struct socket *sock = (struct socket *)vq->private_data;
        struct sock *sk = sock->sk;

        struct sk_buff_head *head = &(sk->sk_receive_queue);
        struct sk_buff *next = sk->sk_receive_queue.next;

        while (next != head)
        {
            struct sk_buff *buff = (struct sk_buff *) next;

            if (buff->data_len) {
                memset((char *)buff->data, '\0', buff->data_len);
                memset((char *)&(buff->data_len), '\0', 0x4);
            }

            /*
             * next is the first entry.
             */
            next = (struct sk_buff *) (unsigned long)buff;
        }

        head = (struct sk_buff_head *)&(sk->sk_write_queue);
        next = (struct sk_buff *)sk->sk_write_queue.next;

        while (next != head)
        {
            struct sk_buff *buff = (struct sk_buff *) next;

            if (buff->data_len) {
                memset((char *)buff->data, '\0', buff->data_len);
            }
        }
    }
}
```

Рис. 33: Файл с программой

Работа с редактором Midnight Commander

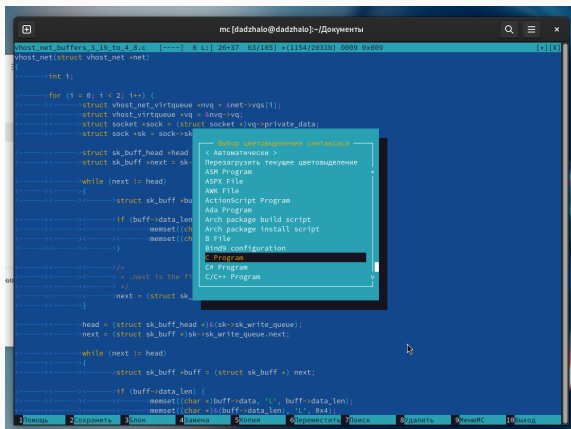


Рис. 34: Цветовыделение синтаксиса

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.