

# Командная оболочка Midnight Commander

---

Амуничников Антон НПИбд-01-22<sup>1</sup>

24 марта, 2023, 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

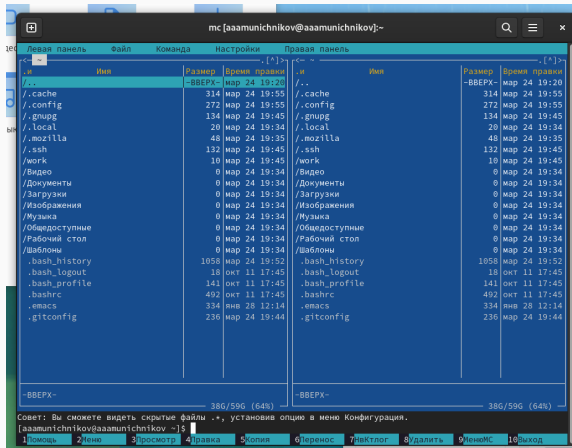


Рис. 1: Запуск mc

# Работа с Midnight Commander

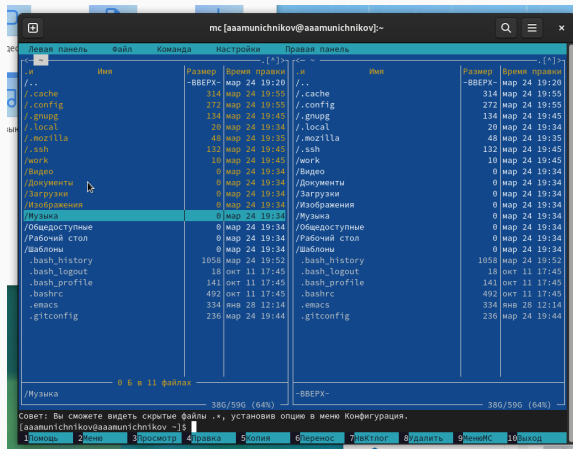


Рис. 2: Выделение

# Работа с Midnight Commander

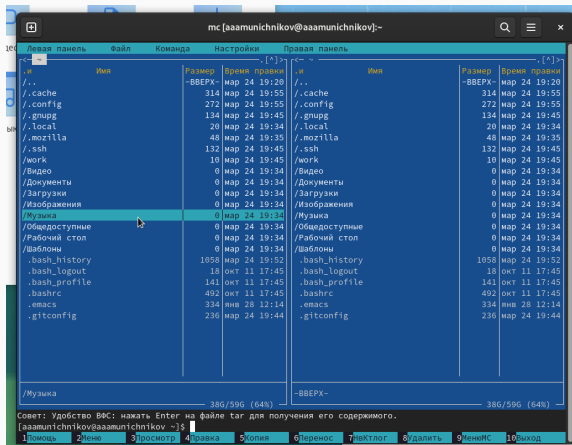


Рис. 3: Отмена



# Работа с Midnight Commander

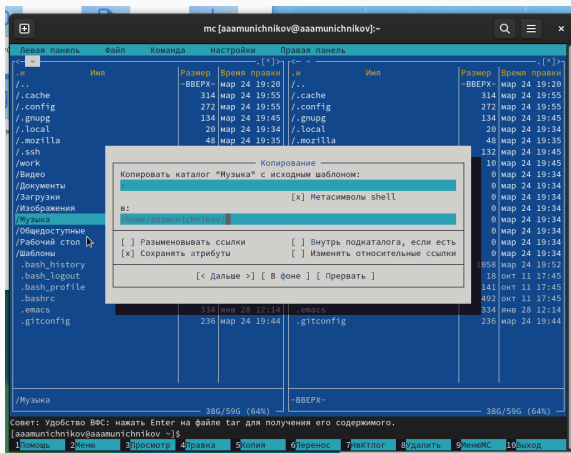


Рис. 4: Копирование

# Работа с Midnight Commander

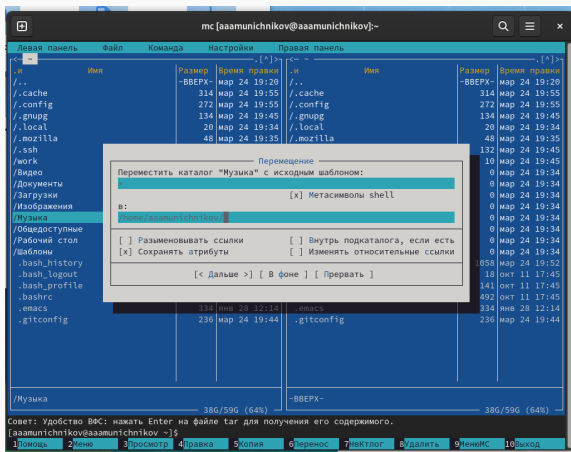


Рис. 5: Перемещение

# Работа с Midnight Commander

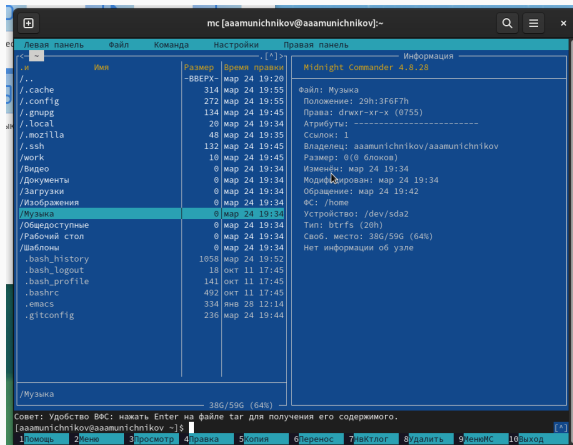


Рис. 6: Информация

# Работа с Midnight Commander

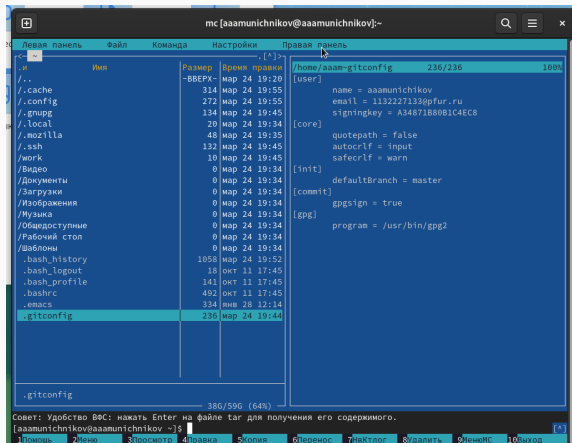


Рис. 7: Быстрый просмотр

# Работа с Midnight Commander

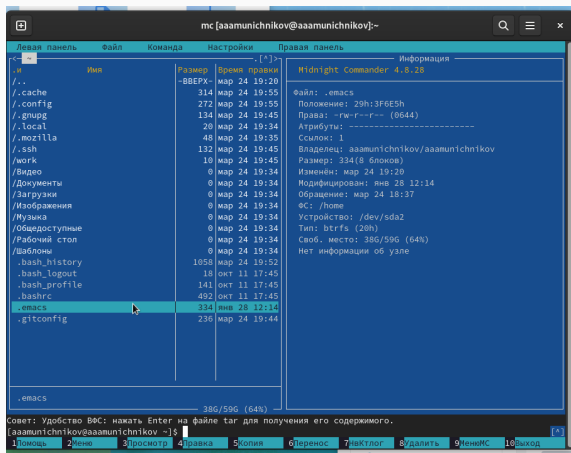


Рис. 8: Информация

# Работа с Midnight Commander

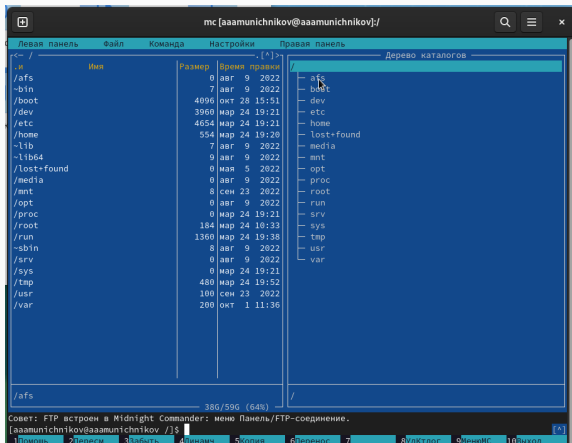
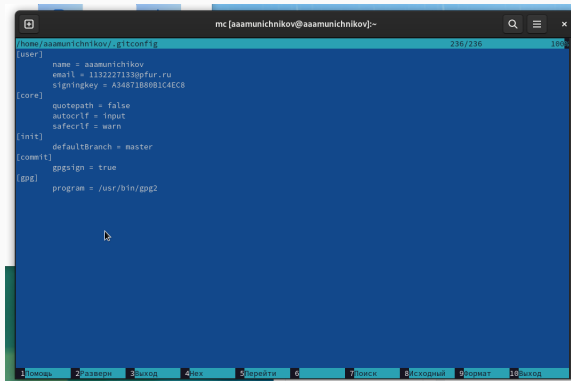


Рис. 9: Дерево каталогов

# Работа с Midnight Commander



The image shows a screenshot of the Midnight Commander (MC) file manager interface. The title bar at the top indicates the user is 'mc [aaamunichnikov@aaamunichnikov]:~'. The main window displays the contents of the file `/home/aaamunichnikov/.gitconfig`. The file's size is shown as 236/236 bytes, and the cursor is positioned at line 180. The file's content is as follows:

```
[user]
  name = aaamunichnikov
  email = 1132227133@pfur.ru
  signingkey = A34871B80B1C4EC8

[core]
  quotepath = false
  autocrlf = input
  safecrlf = warn

[init]
  defaultBranch = master

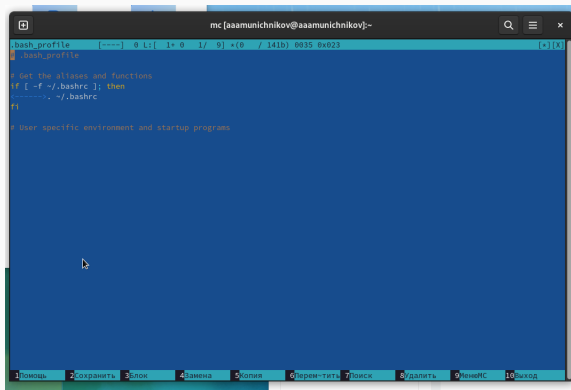
[commit]
  gpgsign = true

[pgp]
  program = /usr/bin/gpg2
```

At the bottom of the window, there is a menu bar with the following items: 1. Помощь (Help), 2. Развернуть (Expand), 3. Выход (Exit), 4. Нех (Not), 5. Перейти (Go), 6. Поиск (Search), 7. Исходный (Original), 8. Формат (Format), 9. Выход (Exit).

Рис. 10: Просмотр содержимого текстового файла

# Работа с Midnight Commander



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



# Работа с Midnight Commander

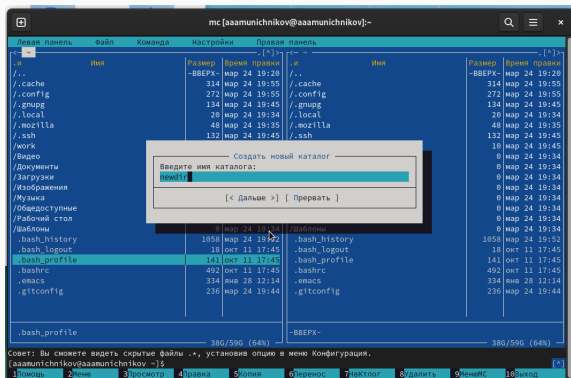


Рис. 12: Создание каталога

# Работа с Midnight Commander

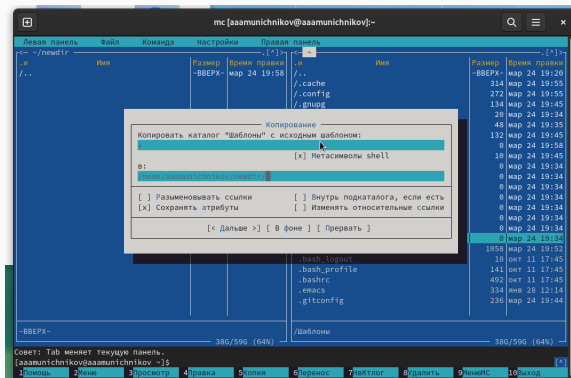


Рис. 13: Копирование в файлов в созданный каталог

# Работа с Midnight Commander

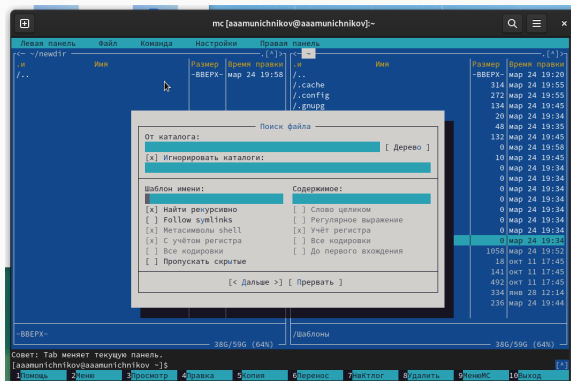


Рис. 14: Поиск файлов

# Работа с Midnight Commander

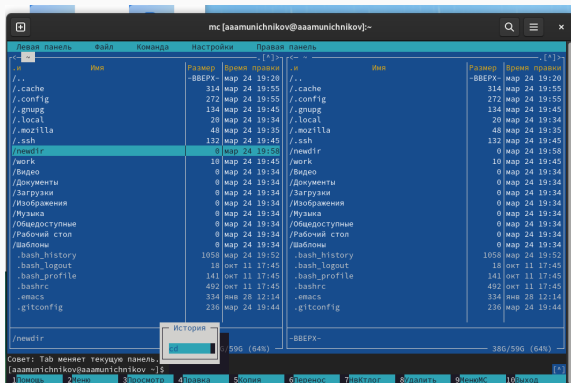


Рис. 15: История команд

# Работа с Midnight Commander

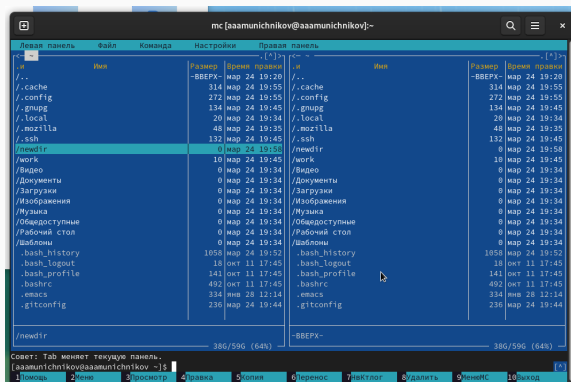
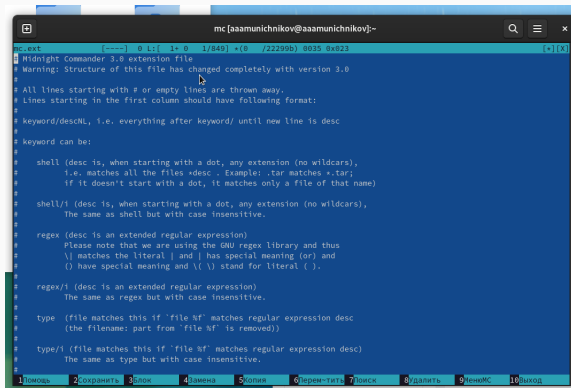


Рис. 16: Переход в домашний каталог

# Работа с Midnight Commander



```
mc.ext      [-----]  0 L: [ 1+ 0  1/849] + (0  /22299b) 0035 0x023  [*)(X)]
# Midnight Commander 3.0 extension file
# Warning: Structure of this file has changed completely with version 3.0
#
# All lines starting with # or empty lines are thrown away.
# Lines starting in the first column should have following format:
#
# keyword/descNL, i.e. everything after keyword/ until new line is desc
#
# keyword can be:
#
#   shell (desc is, when starting with a dot, any extension (no wildcars),
#         i.e. matches all the files <desc> . Example: .tar matches *.tar;
#         If it doesn't start with a dot, it matches only a file of that name)
#
#   shell/! (desc is, when starting with a dot, any extension (no wildcars),
#         The same as shell but with case insensitive.
#
#   regex (desc is an extended regular expression)
#         Please note that we are using the GNU regex library and thus
#         \| matches the literal | and | has special meaning (or) and
#         () have special meaning and \( \) stand for literal ( ).
#
#   regex/! (desc is an extended regular expression)
#         The same as regex but with case insensitive.
#
#   type (file matches this if 'file %f' matches regular expression desc
#         (the filename: part from 'file %f' is removed))
#
#   type/! (file matches this if 'file %f' matches regular expression desc)
#         The same as type but with case insensitive.
#
```

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

Рис. 17: Просмотр файла расширений



# Работа с Midnight Commander

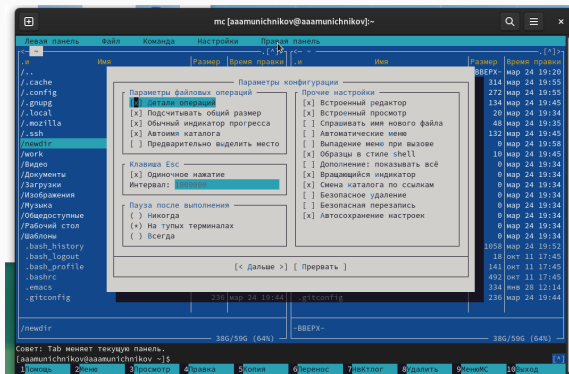


Рис. 19: Конфигурация



# Работа с Midnight Commander

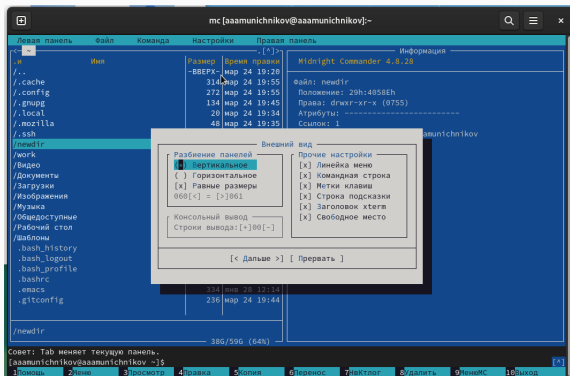


Рис. 20: Внешний вид

# Работа с Midnight Commander

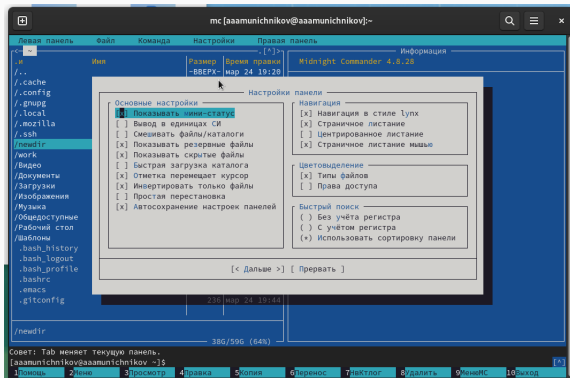


Рис. 21: Настройки панелей

# Работа с Midnight Commander

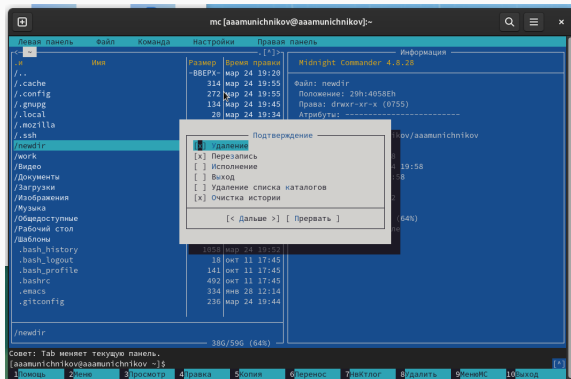


Рис. 22: Подтверждение

# Работа с Midnight Commander

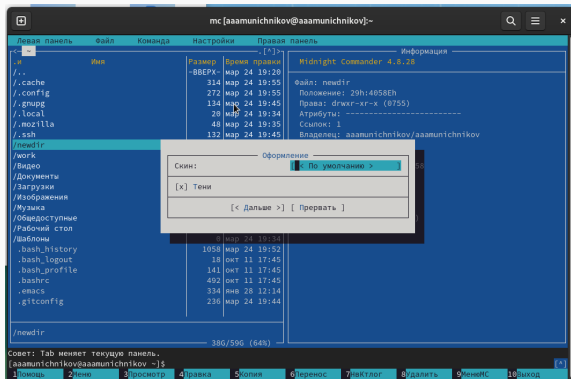


Рис. 23: Оформление

# Работа с Midnight Commander

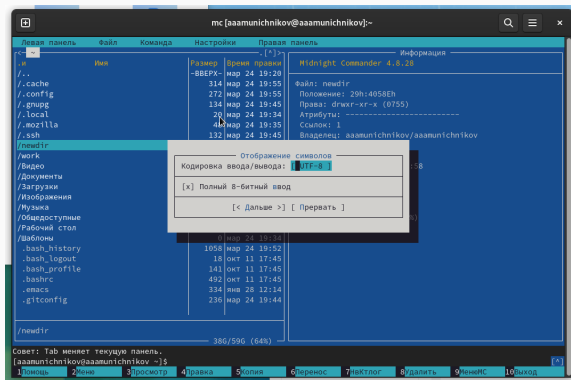


Рис. 24: Кодировка символов

# Работа с Midnight Commander

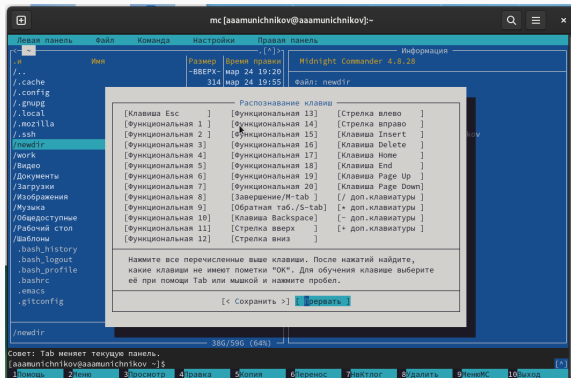


Рис. 25: Распознавание клавиш

# Работа с редактором Midnight Commander

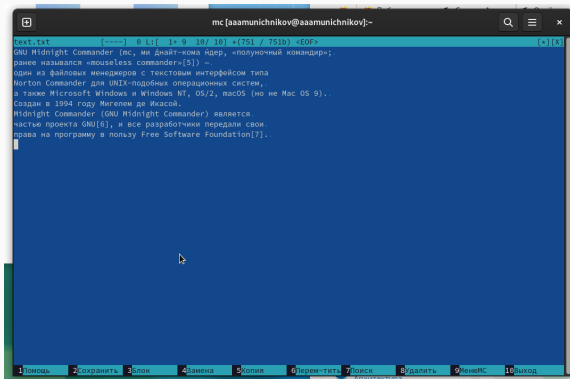


Рис. 26: Файл с текстом

# Работа с редактором Midnight Commander

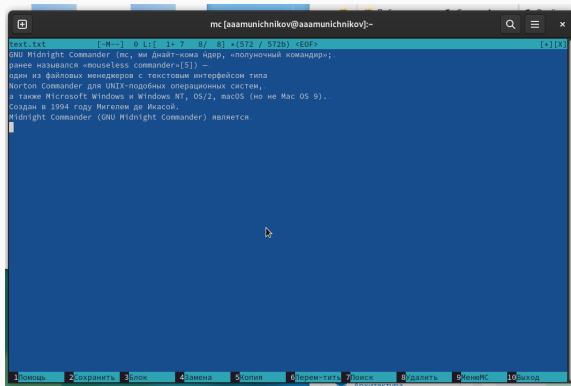


Рис. 27: Файл с текстом



# Работа с редактором Midnight Commander

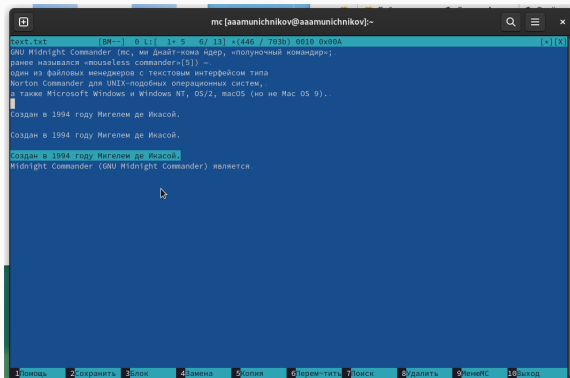


Рис. 28: Копирование фрагмента

# Работа с редактором Midnight Commander

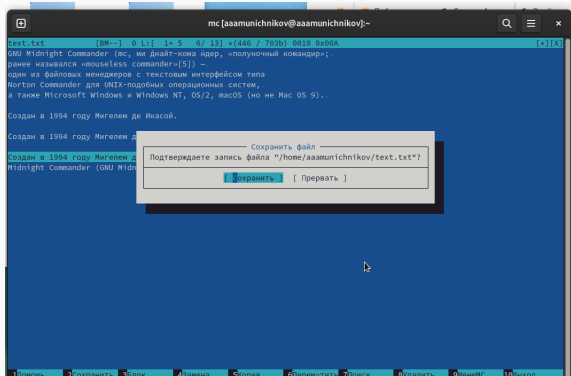


Рис. 29: Сохранение

# Работа с редактором Midnight Commander

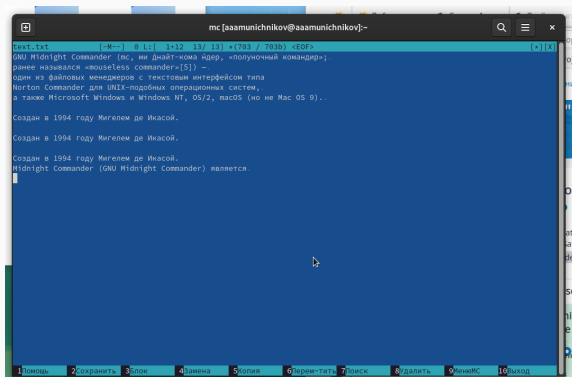


Рис. 30: Отмена

# Работа с редактором Midnight Commander

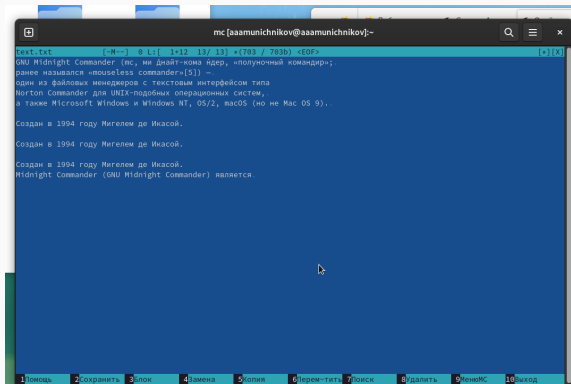


Рис. 31: Переход в конец файла

# Работа с редактором Midnight Commander

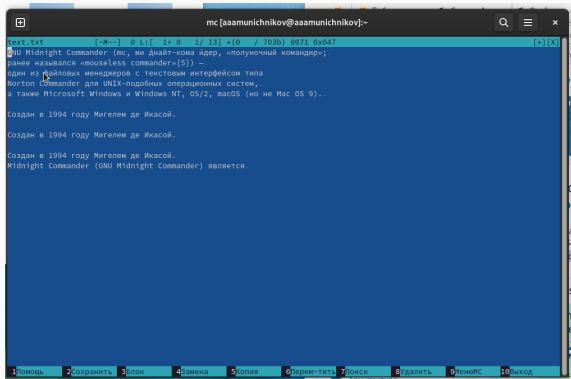
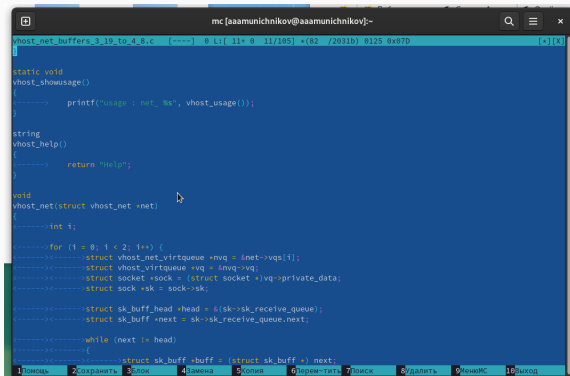


Рис. 32: Переход в начало файла

# Работа с редактором Midnight Commander



```
mc [aaamunichnikov@aaamunichnikov:~  
vhost_net_buffers_3_19_to_4_0.c  (----)  0 L: 11* 0 11/105] + (82 /2031b) 8125 0x07D (*) [X]  
  
static void  
vhost_showusage()  
{  
    printf("usage : net_ %s", vhost_usage());  
}  
  
string  
vhost_help()  
{  
    return "Help";  
}  
  
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;  
            next = next->next;  
        }  
    }  
}
```

Рис. 33: Файл с программой

# Работа с редактором Midnight Commander

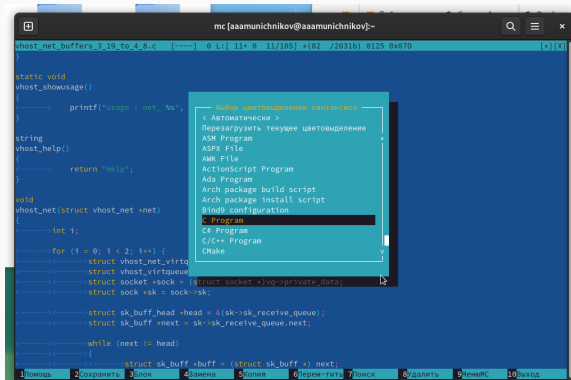


Рис. 34: Цветовыделение синтаксиса

## **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.