

Лабораторная работа №7

Командная оболочка Midnight Commander..

Шуплецов А. А.

25 марта 2023

Российский университет дружбы народов, Москва, Россия

Информация

- Шуплецов Александр Андреевич
- студент ФФМиЕН
- Российский университет дружбы народов
- <https://github.com/winnralex>

Цели и задачи

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

Выполнение работы

Изучим информацию о mc, вызвав в командной строке man mc.

```
aashuplevcov@aashuplevcov:~ — man mc
MC(1)                                     GNU Midnight Commander                               MC(1)

НАИМЕНОВАНИЕ
mc — Визуальная оболочка для Unix-подобных систем.

СИНТАКСИС
mc [-abcCdfhPstuUVx] [-l журнал] [каталог1 [каталог2]] [-e [файл] ...] [-v файл]

ОПИСАНИЕ
Что такое Midnight Commander

Midnight Commander — это программа, предназначенная для просмотра содержимого каталогов и выполнения основных функций управления файлами в UNIX-подобных операционных системах.

КЛЮЧИ
Ключи запуска программы Midnight Commander

-a, --stickchars
Отключить использование графических символов для рисования линий.

-b, --nocolor
Запуск программы в черно-белом режиме экрана.

-c, --color
Включает цветной режим дисплея. Для получения более подробной информации смотрите раздел Цвета.

-c arg, --colors=arg
Используется для того, чтобы задать другой набор цветов в командной строке. Формат аргумента arg описан в разделе Цвета.

--configure-options
Выводит опции конфигурирования, с которыми был собран Midnight Commander.

-d, --nomouse
Отключает поддержку мыши.

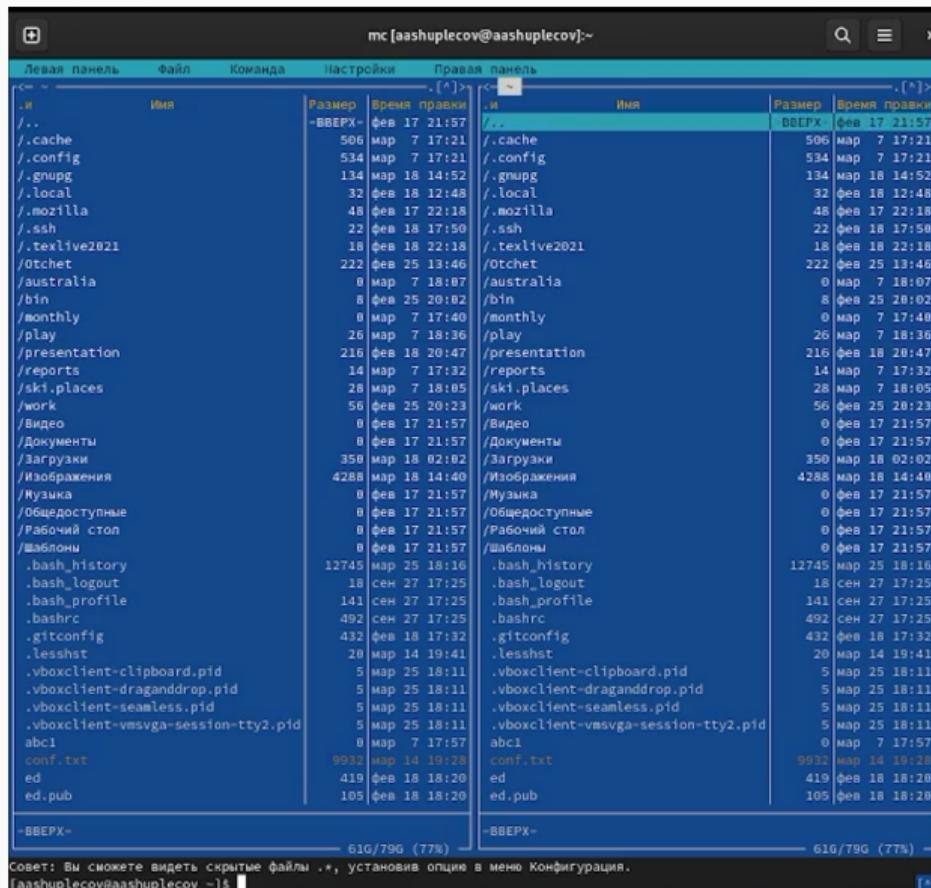
-e \[файл\]
Запустить встроенный редактор. Если параметр "файл" задан, этот файл будет открыт при запуске. Смотрите также mcedit(1).

-f, --datadir
Выводит на экран определённый в процессе компиляции программы путь к файлам программы Midnight Commander.

-F, --datadir=info
Выводит на экран расширенную информацию о путях, используемых при работе программы Midnight Commander.

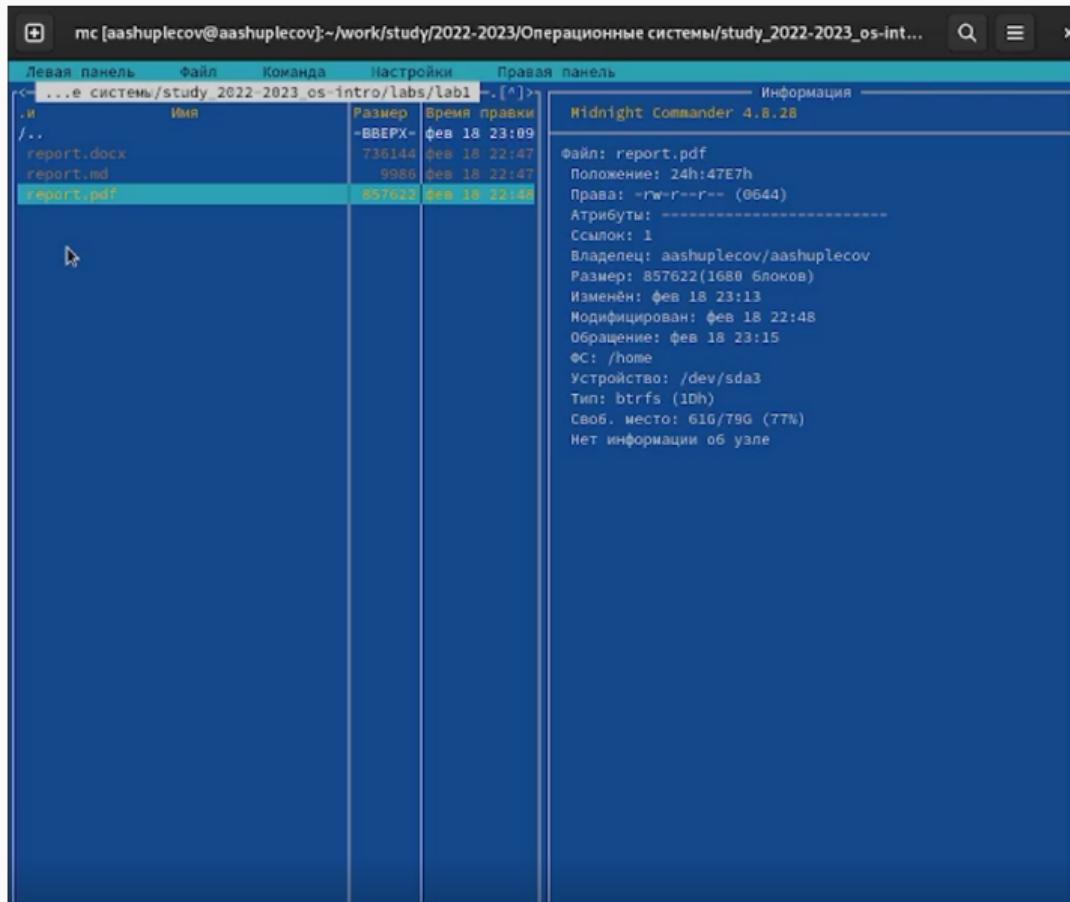
Manual page mc(1) line 1 (press h for help or q to quit)
```

Запустим из командной строки mc, изучим его структуру и меню.

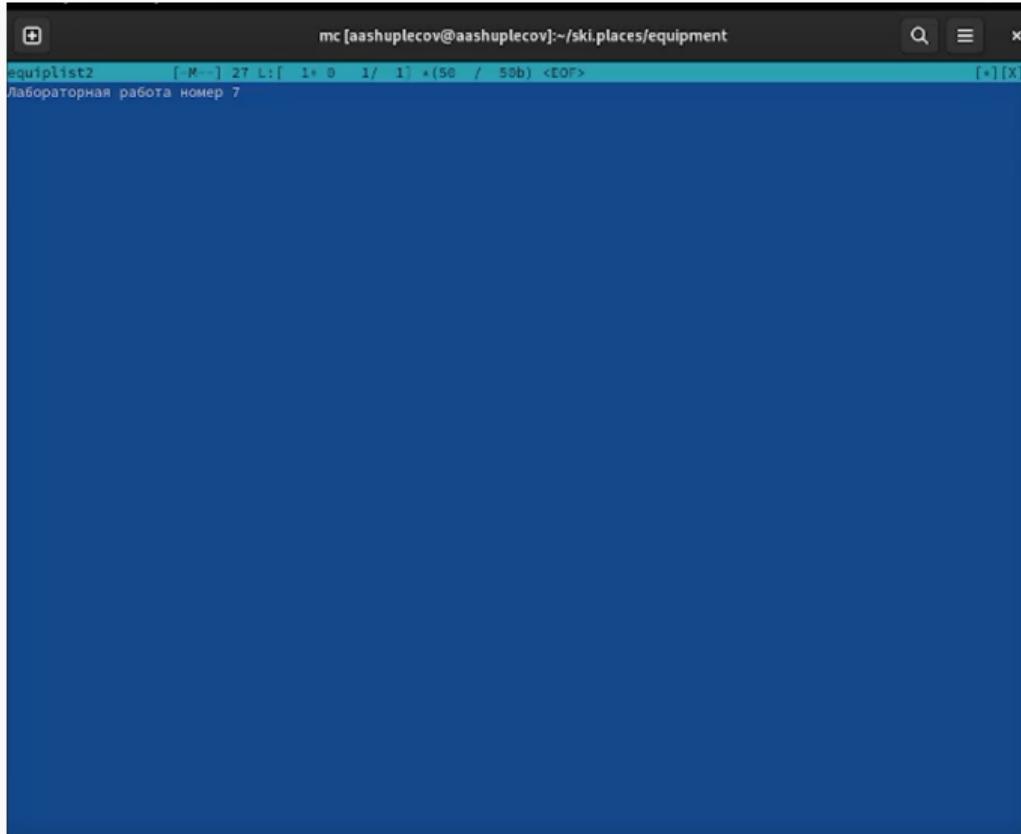


Используя возможности подменю Файл , выполним:

просмотр содержимого текстового файла



редактирование содержимого текстового файла (без сохранения результатов редактирования)



```
mc [aashuplecov@aashuplecov]:~/ski.places/equipment
equiplist2      [ M-- ] 27 L:[ 1* 0  1/  1] ×(50   /  50b) <EOF>
[ * ] [ X ]
Лабораторная работа номер 7
```

создание каталога

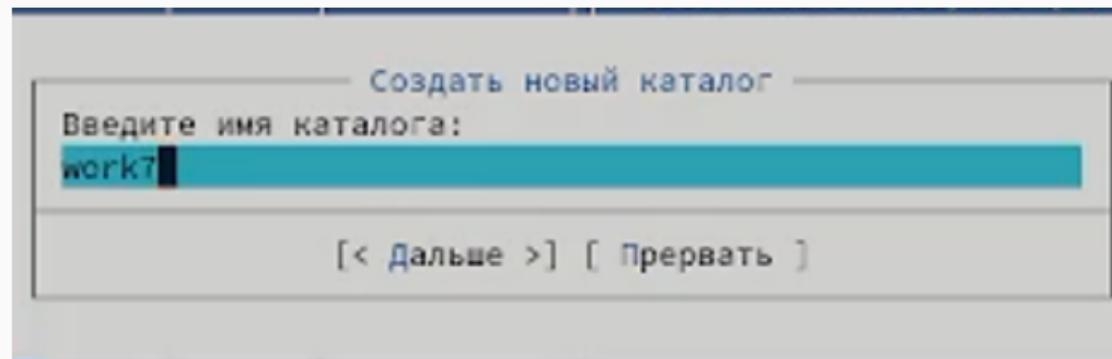


Рис. 5: создание каталога

копирование файлов в созданный каталог

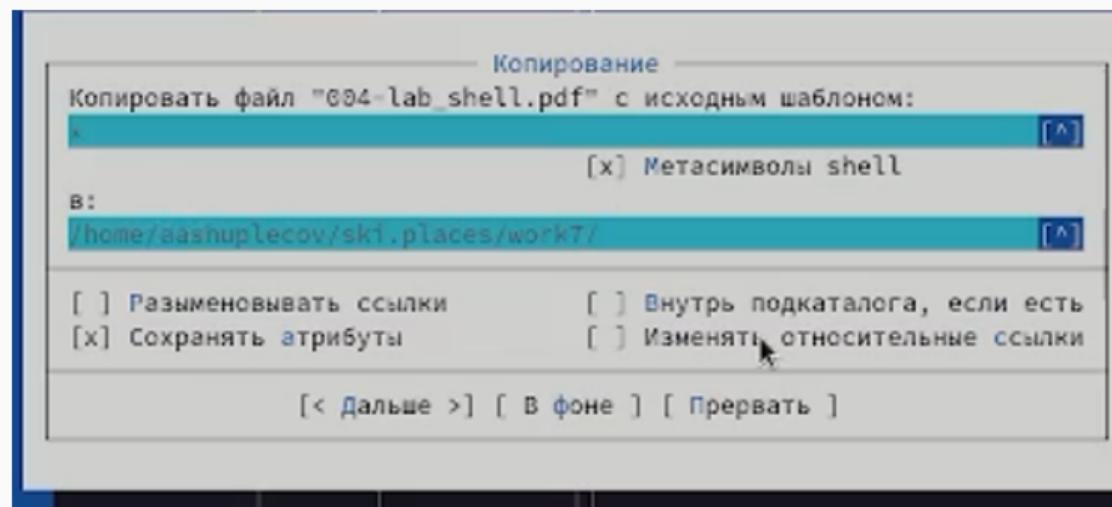
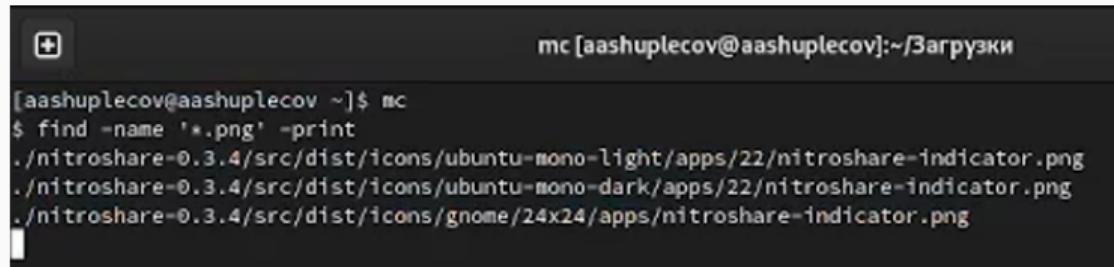


Рис. 6: копирование файлов в созданный каталог

С помощью соответствующих средств подменю Команда осуществим:

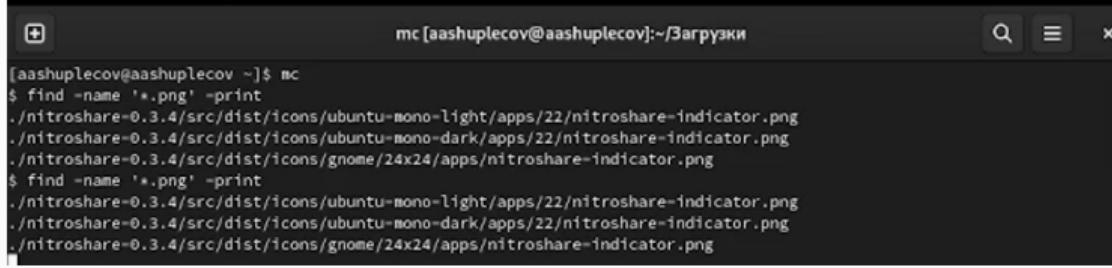
поиск в файловой системе файла с заданными условиями



```
[aashuplecov@aashuplecov ~]$ mc  
mc [aashuplecov@aashuplecov]:~/Загрузки  
[aashuplecov@aashuplecov ~]$ mc  
$ find -name '*.png' -print  
. ./nitroshare=0.3.4/src/dist/icons/ubuntu-mono-light/apps/22/nitroshare=indicator.png  
. ./nitroshare=0.3.4/src/dist/icons/ubuntu-mono-dark/apps/22/nitroshare=indicator.png  
. ./nitroshare=0.3.4/src/dist/icons/gnome/24x24/apps/nitroshare=indicator.png
```

Рис. 7: поиск в файловой системе

выбор и повторение предыдущей команды



The screenshot shows a terminal window titled "mc [aashuplecov@aashuplecov]:~/Загрузки". The window contains the following command history:

```
[aashuplecov@aashuplecov ~]$ mc
$ find -name '*.png' -print
./nitroshare=0.3.4/src/dist/icons/ubuntu-mono-light/apps/22/nitroshare=indicator.png
./nitroshare=0.3.4/src/dist/icons/ubuntu-mono-dark/apps/22/nitroshare=indicator.png
./nitroshare=0.3.4/src/dist/icons/gnome/24x24/apps/nitroshare=indicator.png
$ find -name '*.png' -print
./nitroshare=0.3.4/src/dist/icons/ubuntu-mono-light/apps/22/nitroshare=indicator.png
./nitroshare=0.3.4/src/dist/icons/ubuntu-mono-dark/apps/22/nitroshare=indicator.png
./nitroshare=0.3.4/src/dist/icons/gnome/24x24/apps/nitroshare=indicator.png
```

Рис. 8: выбор и повторение предыдущей команды

анализ файла меню

```
mc [aashuplecov@aashuplecov]:~
```

```
menu.mmc      [----] 48 L:[ 1=17 18/370] *(781 /11820b) 6097 0x061
```

```
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 Run 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
```

```
%if ditto
```

```
%of 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
```

```
0
```

```
Edit a bug report and send it to root
```

```
I='mktemp "$(MC_TMPDIR=/tmp)/mail.XXXXXX"' || exit 1
```

```
$(EDITOR-v1) "$I"
```

```
test -r "$I" && mail root < "$I"
```

```
rm -f "$I"
```

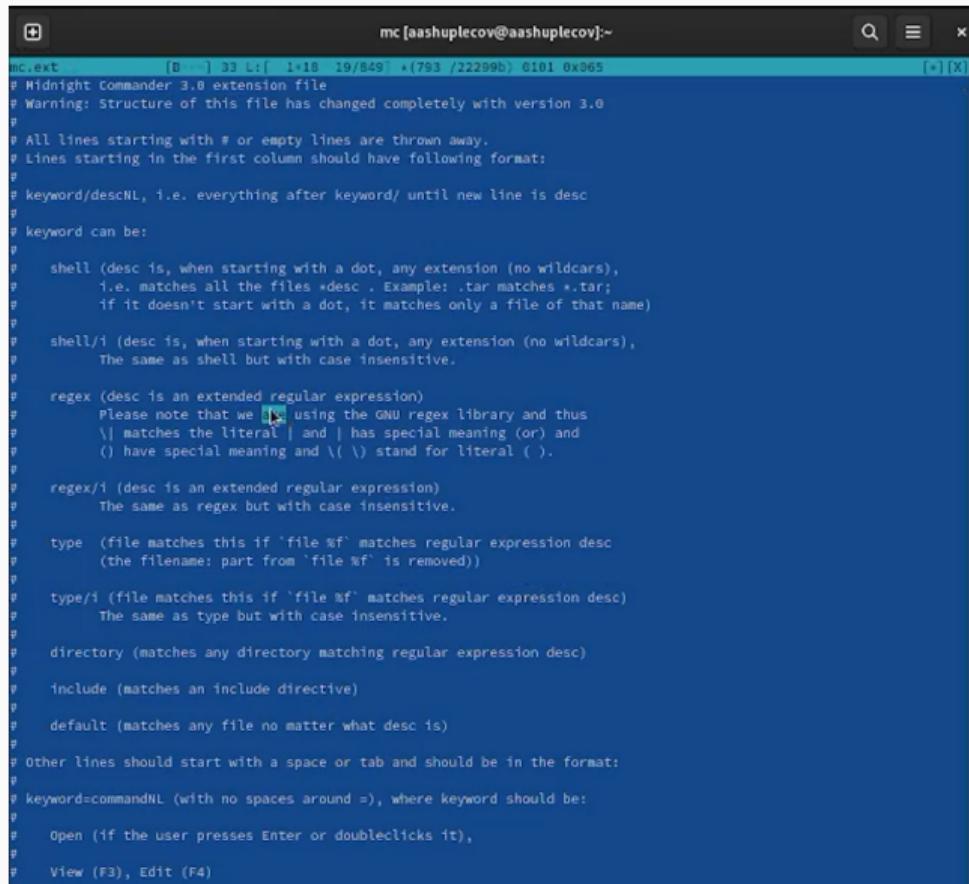
```
=> f \.16 | f \.38 | f \.45 | f \.56 | f \.65 | f \.76 | f \.85 | f \.mans & t r
```

```
l
```

```
Display the file with roff -man
```

```
%view(ascii,nroff) roff -c -Tlatin1 -mandoc %f
```

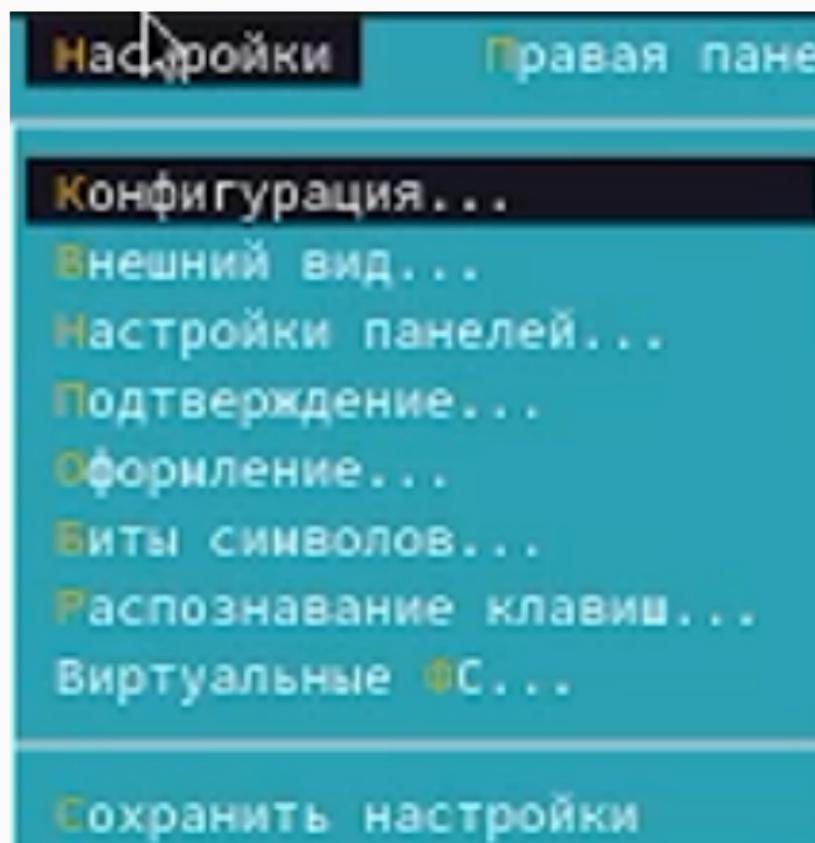
анализ файла расширений



mc [aashuplecov@aashuplecov:~]

```
mc.ext          [0 ..] 33 L:[ 1+18 19/B49 ] *(793 /22299b) 6101 0x065
# 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/i (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/i (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/i (file matches this if 'file %f' matches regular expression desc)
# The same as type but with case insensitive.
#
# directory (matches any directory matching regular expression desc)
#
# include (matches an include directive)
#
# default (matches any file no matter what desc is)
#
# Other lines should start with a space or tab and should be in the format:
#
# keyword=commandNL (with no spaces around =), where keyword should be:
#
# Open (if the user presses Enter or doubleclicks it),
#
# View (F3), Edit (F4)
```

Вызовем подменю Настройки. Освоим операции, определяющие структуру экрана mc (Full screen, Double Width, Show Hidden Files и т.д.).

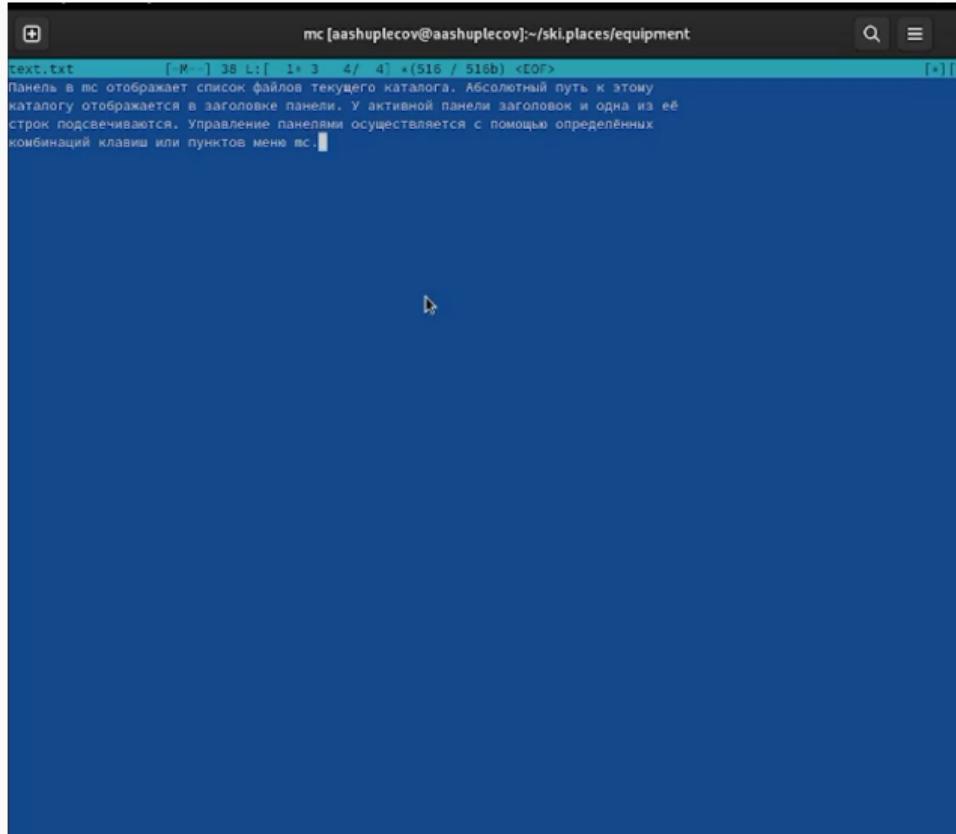


Создадим текстовой файл text.txt

<= ~/ski.places/equipment		[^]>	
.И	Имя	Размер	Время правки
/..		-ВВЕРХ-	мар 25 18:38
	equiplist	5086	мар 7 17:51
	equiplist2	0	мар 7 17:57
	text.txt	0	мар 25 19:05

Рис. 12: создание текстового файла text.txt

Откроем этот файл с помощью встроенного в mc редактора, вставим в открытый файл небольшой фрагмент текста, скопированный из Интернета.



Поработаем с текстовым файлом text.txt и сохраним изменения в нем.

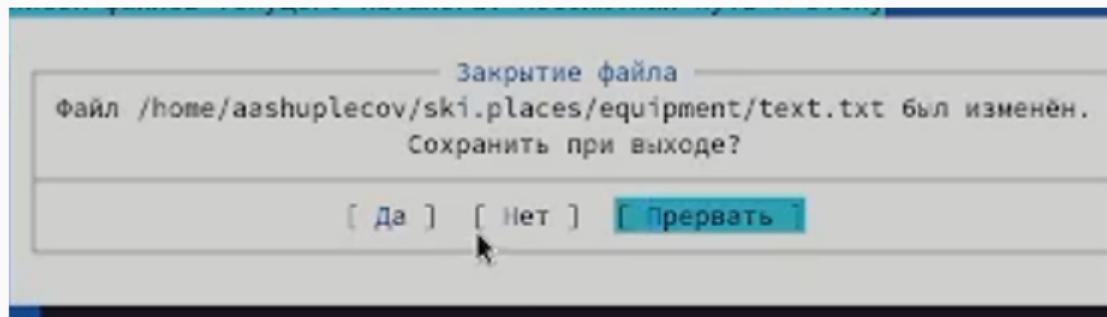
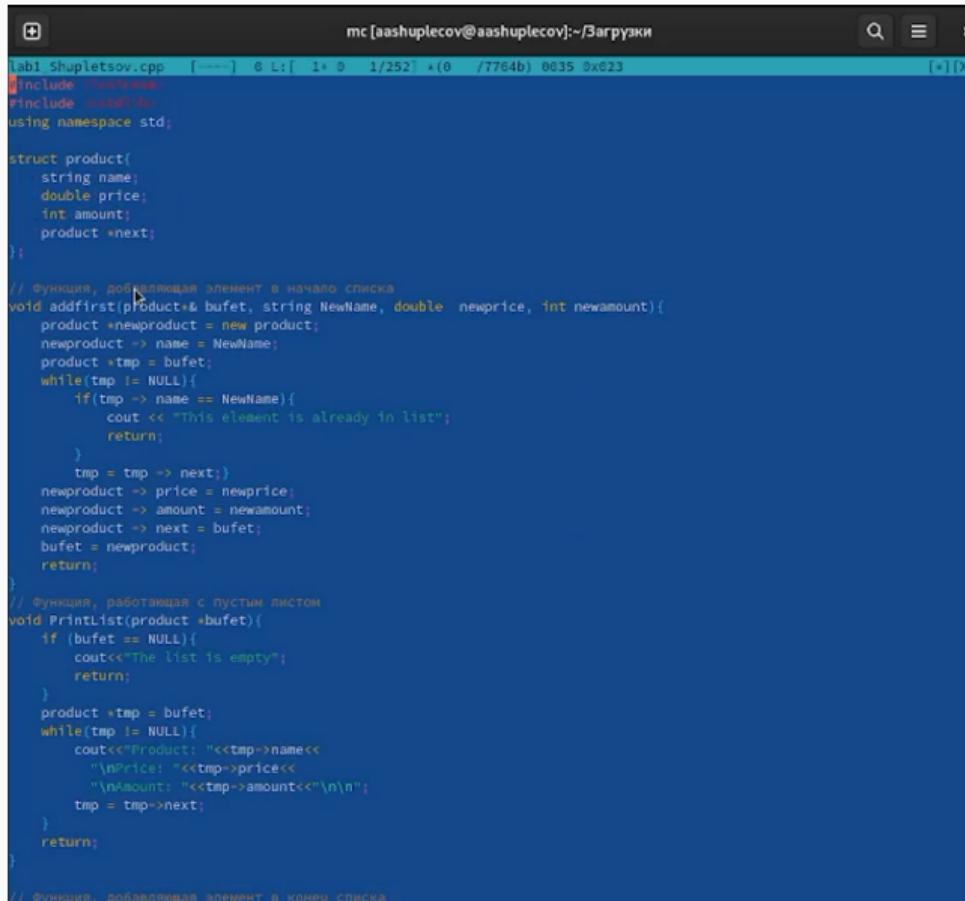


Рис. 14: сохранение изменений в файле text.txt

Откроем файл с исходным текстом на языке программирования C++.



The screenshot shows a terminal window titled "mc [aashupletsov@aashupletsov:~/Загрузки]" containing C++ source code. The code defines a linked list structure named 'product' with fields for name, price, amount, and a pointer to the next node. It includes functions to add elements to the beginning of the list ('addfirst'), print the list ('PrintList'), and add elements to the end of the list ('addlast'). The code uses standard input/output streams and header files <iostream> and <iomanip>. The terminal interface includes a search bar, a menu icon, and a close button.

```
lab1 Shupletsov.cpp  [-----]  0 L:[ 1+ 0   1/252] *(0   /7764b) 0035 0x023
#include <iostream>
#include <iomanip>
using namespace std;

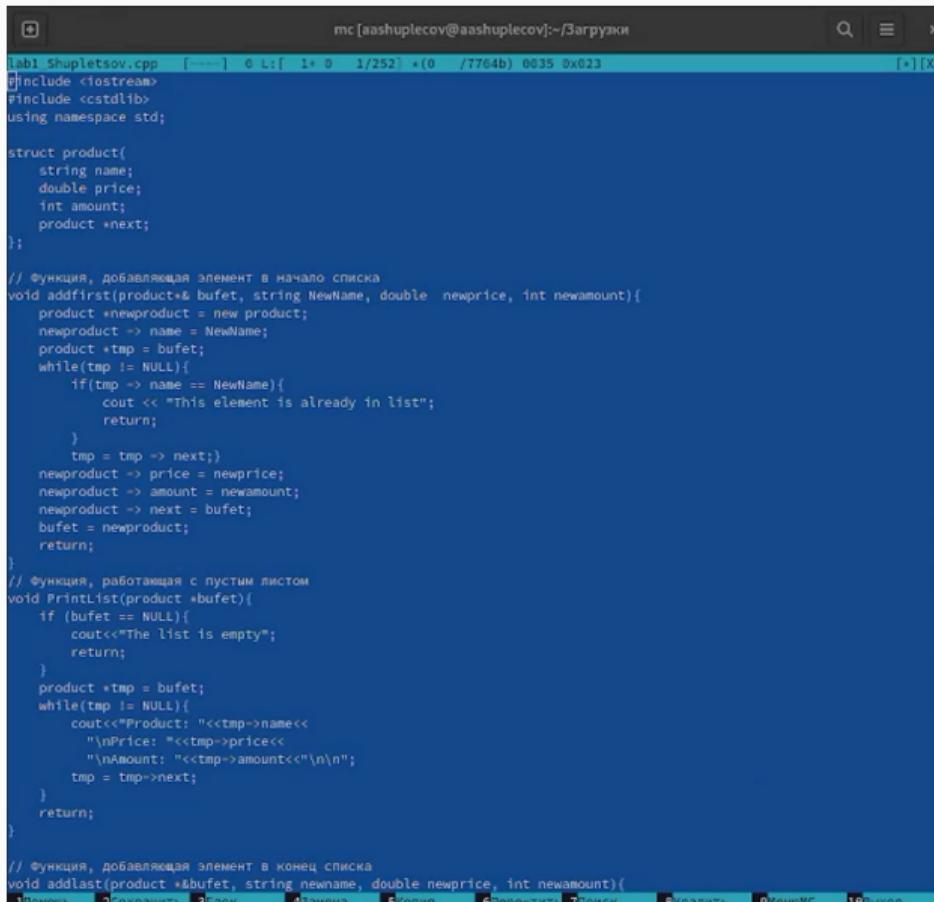
struct product{
    string name;
    double price;
    int amount;
    product *next;
};

// Функция, добавляющая элемент в начало списка
void addfirst(product*& bufet, string NewName, double newprice, int newamount){
    product *newproduct = new product;
    newproduct->name = NewName;
    product *tmp = bufet;
    while(tmp != NULL){
        if(tmp->name == NewName){
            cout << "This element is already in list";
            return;
        }
        tmp = tmp->next;
    }
    newproduct->price = newprice;
    newproduct->amount = newamount;
    newproduct->next = bufet;
    bufet = newproduct;
    return;
}

// Функция, работающая с пустым листом
void PrintList(product *&bufet){
    if (bufet == NULL){
        cout<<"The list is empty";
        return;
    }
    product *tmp = bufet;
    while(tmp != NULL){
        cout<<"Product: "<<tmp->name<<
        "\nPrice: "<<tmp->price<<
        "\nAmount: "<<tmp->amount<<"\n\n";
        tmp = tmp->next;
    }
    return;
}

// Функция, добавляющая элемент в конец списка
```

Используя меню редактора, выключим подсветку синтаксиса.



The screenshot shows a terminal window titled "mc [aashuplevcov@aashuplevcov]:~/Загрузки". The code displayed is a C++ program named "Shupletsov.cpp". The code defines a linked list structure "product" with fields name, price, amount, and a pointer to the next node. It includes functions to add elements at the beginning ("addfirst"), end ("addlast"), and print the list ("PrintList"). The code uses standard C++ headers like <iostream> and <cstdlib>. The terminal window has a dark theme with syntax highlighting for code. The status bar at the bottom shows file names and line numbers.

```
#include <iostream>
#include <cstdlib>
using namespace std;

struct product{
    string name;
    double price;
    int amount;
    product *next;
};

// Функция, добавляющая элемент в начало списка
void addfirst(product*& bufet, string NewName, double newprice, int newamount){
    product *newproduct = new product;
    newproduct -> name = NewName;
    product *tmp = bufet;
    while(tmp != NULL){
        if(tmp -> name == NewName){
            cout << "This element is already in list";
            return;
        }
        tmp = tmp -> next;
    }
    newproduct -> price = newprice;
    newproduct -> amount = newamount;
    newproduct -> next = bufet;
    bufet = newproduct;
    return;
}

// Функция, работающая с пустым листом
void PrintList(product *&bufet){
    if (bufet == NULL){
        cout << "The list is empty";
        return;
    }
    product *tmp = bufet;
    while(tmp != NULL){
        cout << "Product: "<<tmp->name<<
        "\nPrice: "<<tmp->price<<
        "\nAmount: "<<tmp->amount<<"\n\n";
        tmp = tmp->next;
    }
    return;
}

// Функция, добавляющая элемент в конец списка
void addlast(product *&bufet, string newname, double newprice, int newamount){
```

Вывод

Я освоил основные возможности командной оболочки Midnight Commander, приобрел навыки практической работы по просмотру каталогов и файлов, манипуляций с ними.

Список литературы

Кулябов Д.С. “Материалы к лабораторным работам”