

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

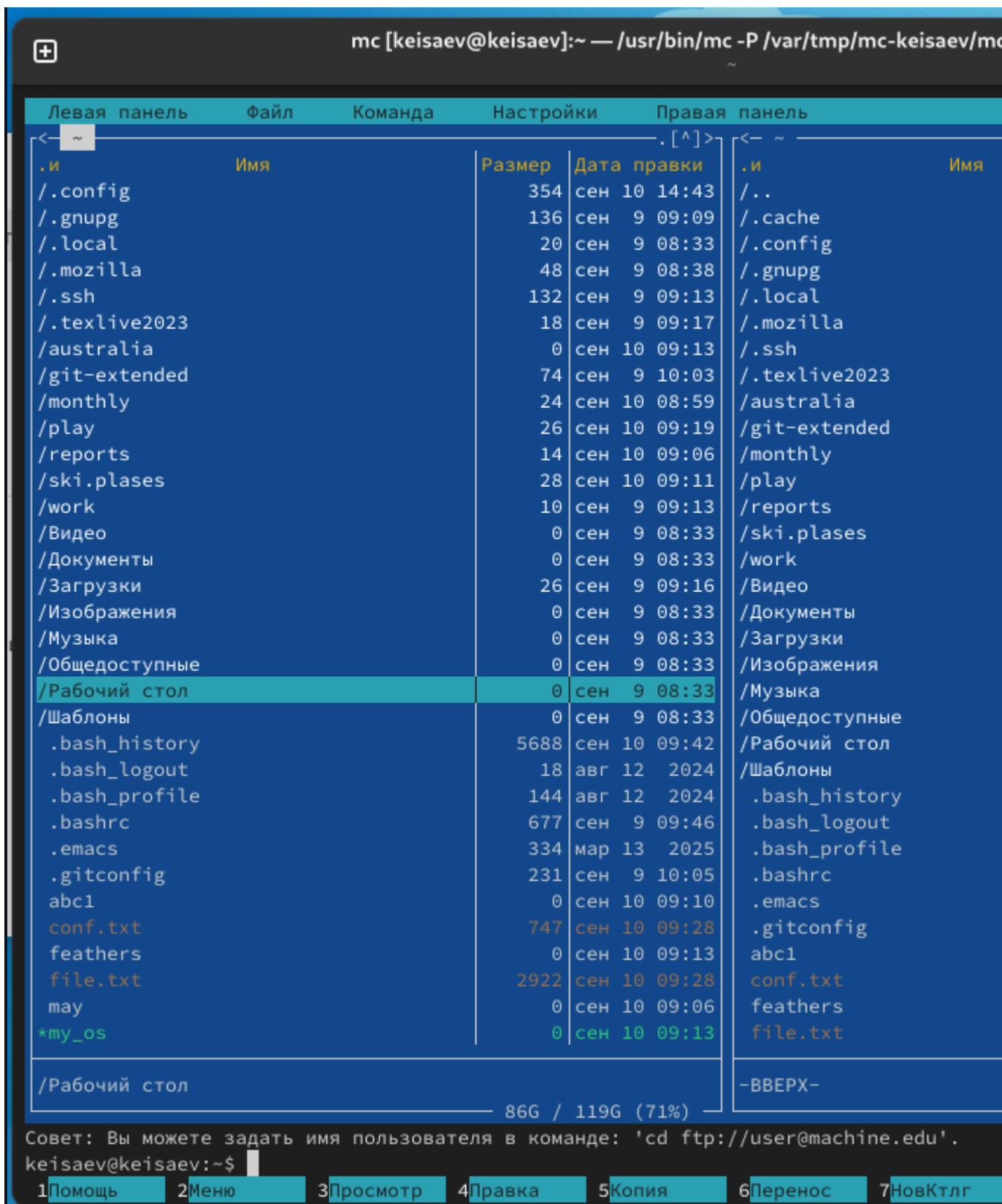
Работа с Midnight Commander

```
mc [keisaev@keisaev]:~ — /usr/bin/mc -P /var/tmp/mc-keisaev/mc

+
Левая панель      Файл      Команда      Настройки      Правая панель
<— ~ — .[^]>
.и      Имя      Размер      Дата правки      .и      Имя
/..      -ВВЕРХ-      сен 9 08:31      /..
/.cache      496      сен 10 14:43      /.cache
/.config      354      сен 10 14:43      /.config
/.gnupg      136      сен 9 09:09      /.gnupg
/.local      20      сен 9 08:33      /.local
/.mozilla      48      сен 9 08:38      /.mozilla
/.ssh      132      сен 9 09:13      /.ssh
/.texlive2023      18      сен 9 09:17      /.texlive2023
/australia      0      сен 10 09:13      /australia
/git-extended      74      сен 9 10:03      /git-extended
/monthly      24      сен 10 08:59      /monthly
/play      26      сен 10 09:19      /play
/reports      14      сен 10 09:06      /reports
/ski.plases      28      сен 10 09:11      /ski.plases
/work      10      сен 9 09:13      /work
/Видео      0      сен 9 08:33      /Видео
/Документы      0      сен 9 08:33      /Документы
/Загрузки      26      сен 9 09:16      /Загрузки
/Изображения      0      сен 9 08:33      /Изображения
/Музыка      0      сен 9 08:33      /Музыка
/Общедоступные      0      сен 9 08:33      /Общедоступные
/Рабочий стол      0      сен 9 08:33      /Рабочий стол
/Шаблоны      0      сен 9 08:33      /Шаблоны
.bash_history      5688      сен 10 09:42      .bash_history
.bash_logout      18      авг 12 2024      .bash_logout
.bash_profile      144      авг 12 2024      .bash_profile
.bashrc      677      сен 9 09:46      .bashrc
.emacs      334      мар 13 2025      .emacs
.gitconfig      231      сен 9 10:05      .gitconfig
abc1      0      сен 10 09:10      abc1
conf.txt      747      сен 10 09:28      conf.txt
feathers      0      сен 10 09:13      feathers
file.txt      2922      сен 10 09:28      file.txt
-ВВЕРХ-
86G / 119G (71%)
Совет: Хотите видеть резервные файлы .~ ? Установите опцию в меню Конфигурация.
keisaev@keisaev:~$
1Помощь 2Меню 3Просмотр 4Правка 5Копия 6Перенос 7НовКтлг
```

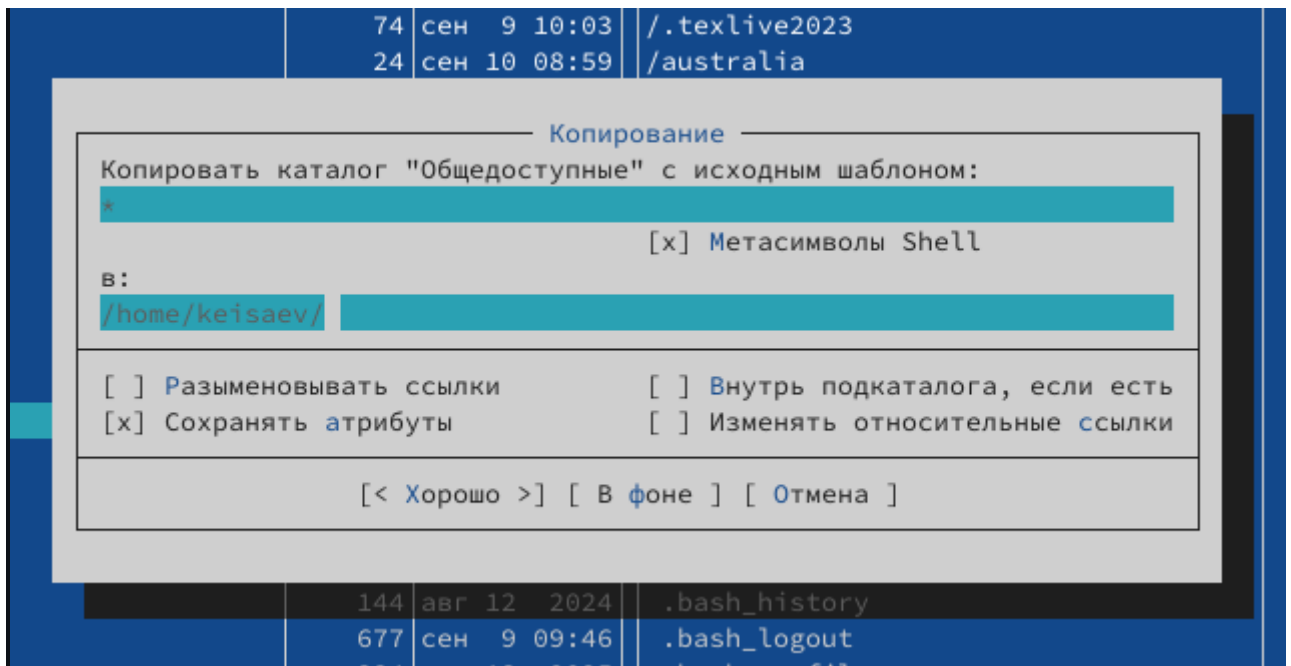
{ #fig:001 width=70% height=70% }

Работа с Midnight Commander



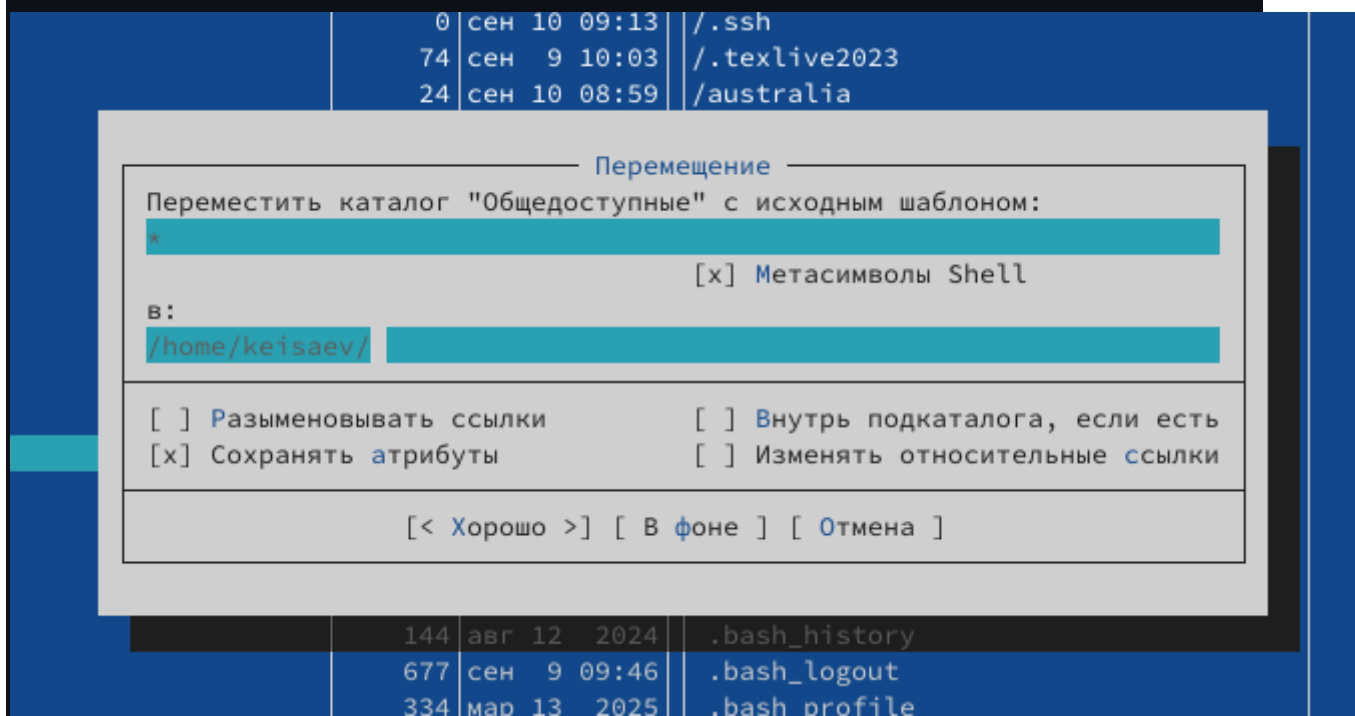
{ #fig:003 width=70% height=70% }

Работа с Midnight Commander



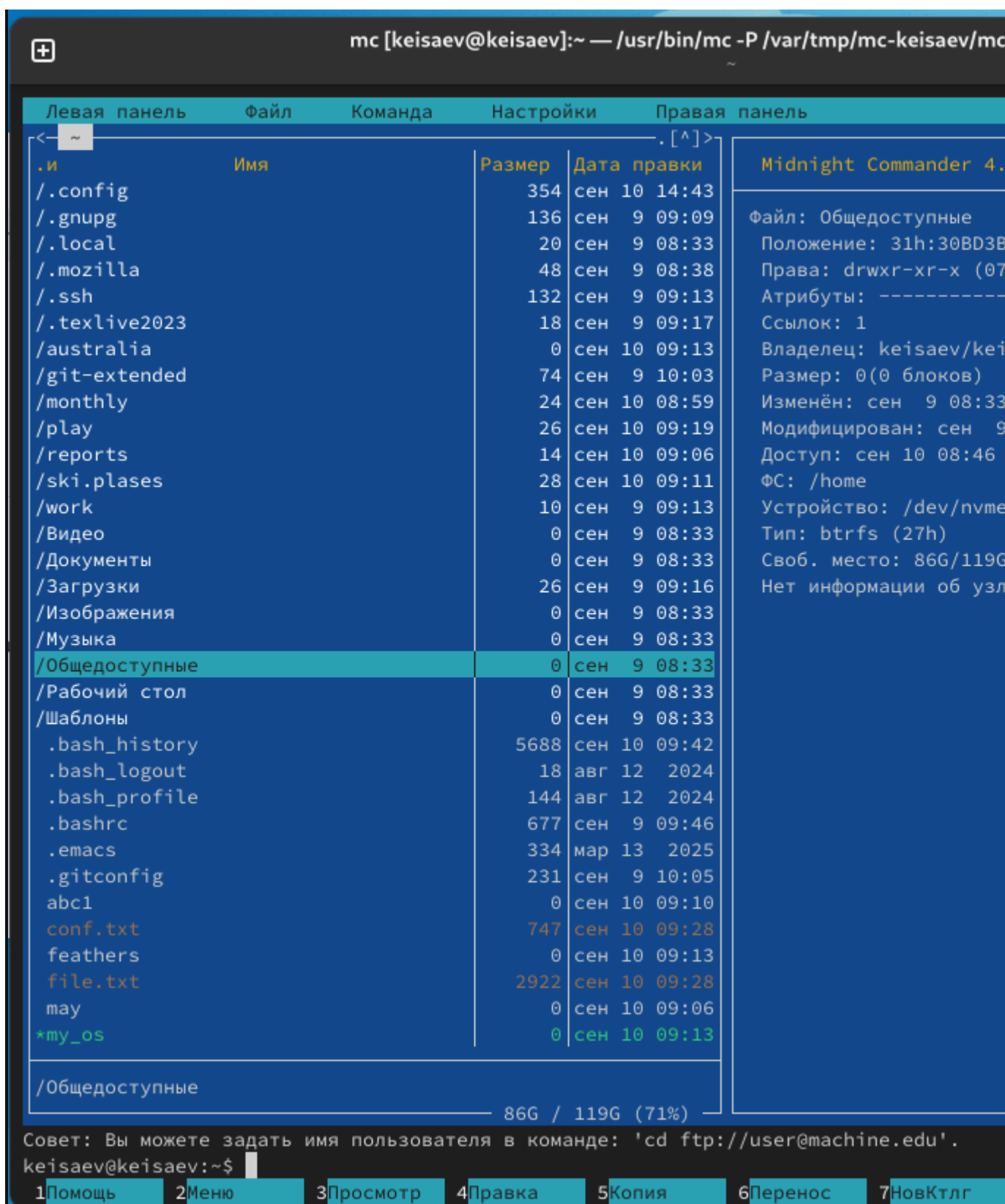
{ #fig:004 width=70% height=70% }

Работа с Midnight Commander



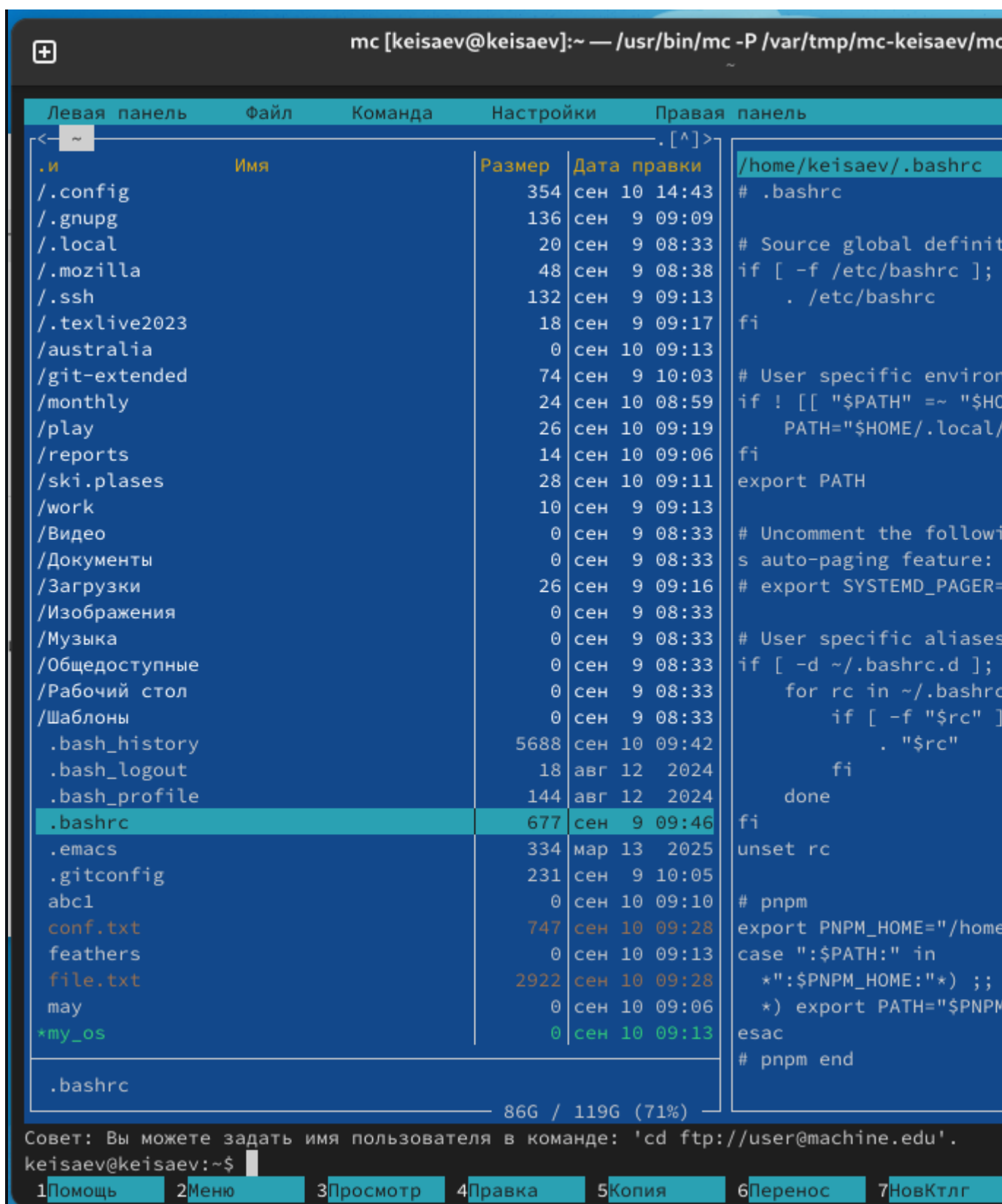
{ #fig:005 width=70% height=70% }

Работа с Midnight Commander



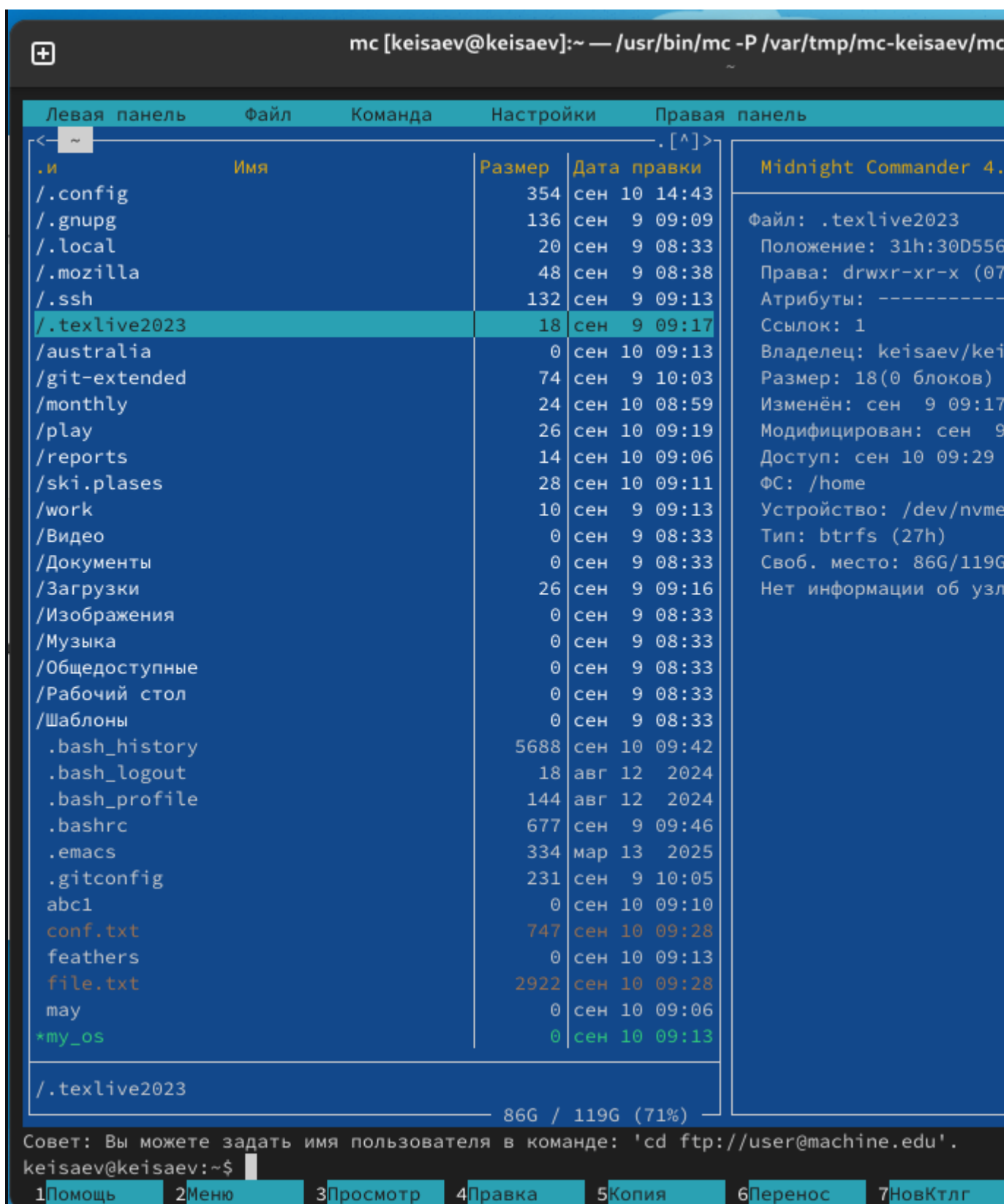
{ #fig:006 width=70% height=70% }

Работа с Midnight Commander



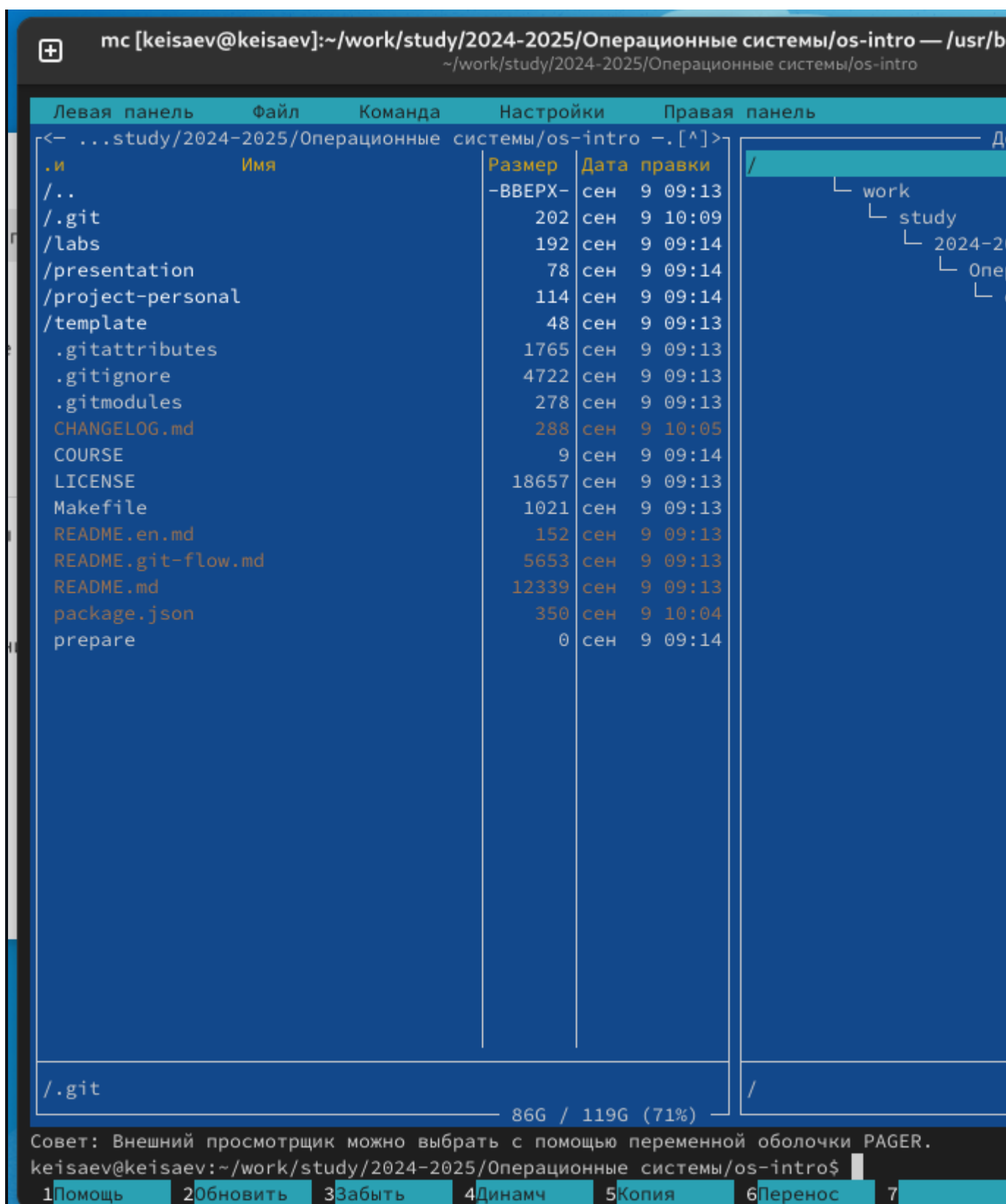
{ #fig:007 width=70% height=70% }

Работа с Midnight Commander



{ #fig:008 width=70% height=70% }

Работа с Midnight Commander



{ #fig:009 width=70% height=70% }

Работа с Midnight Commander

```
mc [keisaev@keisaev]:~/work/study/2024-2025/Операционные системы/os-intro — /usr/bi
~/work/study/2024-2025/Операционные системы/os-intro

/home/keisaev/work/study/2024-2025/Операционные системы/os-intro/Makefile
SHELL := /bin/bash
COURSE =

.PHONY: all clean

all: help

help:
    @echo 'Usage:'
    @echo '  make <target>'
    @echo
    @echo 'Targets:'
    @grep -E '^[a-zA-Z_0-9.-]+:.*?###.*$$' $(MAKEFILE_LIST) | grep -v '###' | sort
:.*?###"; {printf " \033[36m%-30s\033[0m %s\n", $$1, $$2}'
    @grep -E '^###.*' $(MAKEFILE_LIST) | cut -d' ' -f2- | awk 'BEGIN {FS = "###"
    @grep -E '^[a-zA-Z_0-9.-]+:.*?###.*$$' $(MAKEFILE_LIST) | sort | cut -d: -f2
ntf " \033[36m%-30s\033[0m %s\n", $$1, $$2}'
    @echo

list:    ## List of courses
    @. ./template/config/script/list-courses

prepare:    ## Generate directories structure
    @. ./template/config/script/prepare
    @touch prepare

submodule:    ## Update submules
    git submodule update --init --recursive
    git submodule foreach 'git fetch origin; git checkout $$ (git rev-parse --abb
n/$$(git rev-parse --abbrev-ref HEAD); git submodule update --recursive; git clean -
```

1Помощь

2Разверн

3Выход

4Hex

5Перейти

6

7Поиск

{ #fig:010 width=70% height=70% }

Работа с Midnight Commander

```
mc [keisaev@keisaev]:~/work/study/2024-2025/Операционные системы/os-intro — /usr/bi
~/work/study/2024-2025/Операционные системы/os-intro

Makefile      [----]  0 L:[  1+ 0   1/ 28] *(0   /1021b) 0083 0x053
SHELL := /bin/bash
COURSE =.

.PHONY: all clean

all: help

help:
<----->@echo 'Usage:'
<----->@echo '  make <target>'
<----->@echo.
<----->@echo 'Targets:'
<----->@grep -E '^[a-zA-Z_0-9.-]+:.*?###.*$$' $(MAKEFILE_LIST) | grep -v '###' | sort
<----->@grep -E '^###.*' $(MAKEFILE_LIST) | cut -d' ' -f2- | awk 'BEGIN {FS = "###"}
<----->@grep -E '^[a-zA-Z_0-9.-]+:.*?###.*$$' $(MAKEFILE_LIST) | sort | cut -d: -f2
<----->@echo

list:<->## List of courses
<----->@. ./template/config/script/list-courses

prepare:<----->## Generate directories structure
<----->@. ./template/config/script/prepare
<----->@touch prepare

submodule:<----->## Update submules
<----->git submodule update --init --recursive
<----->git submodule foreach 'git fetch origin; git checkout $$ (git rev-parse --abb
```

1Помощь

2Сохран

3Блок

4Замена

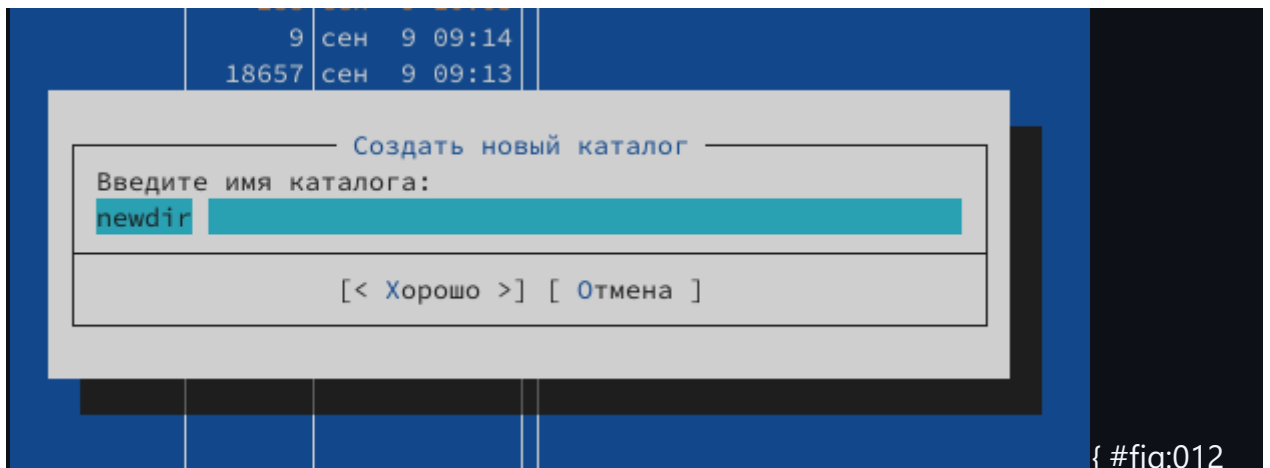
5Копия

6Перем~тить

7Поиск

{ #fig:011 width=70% height=70% }

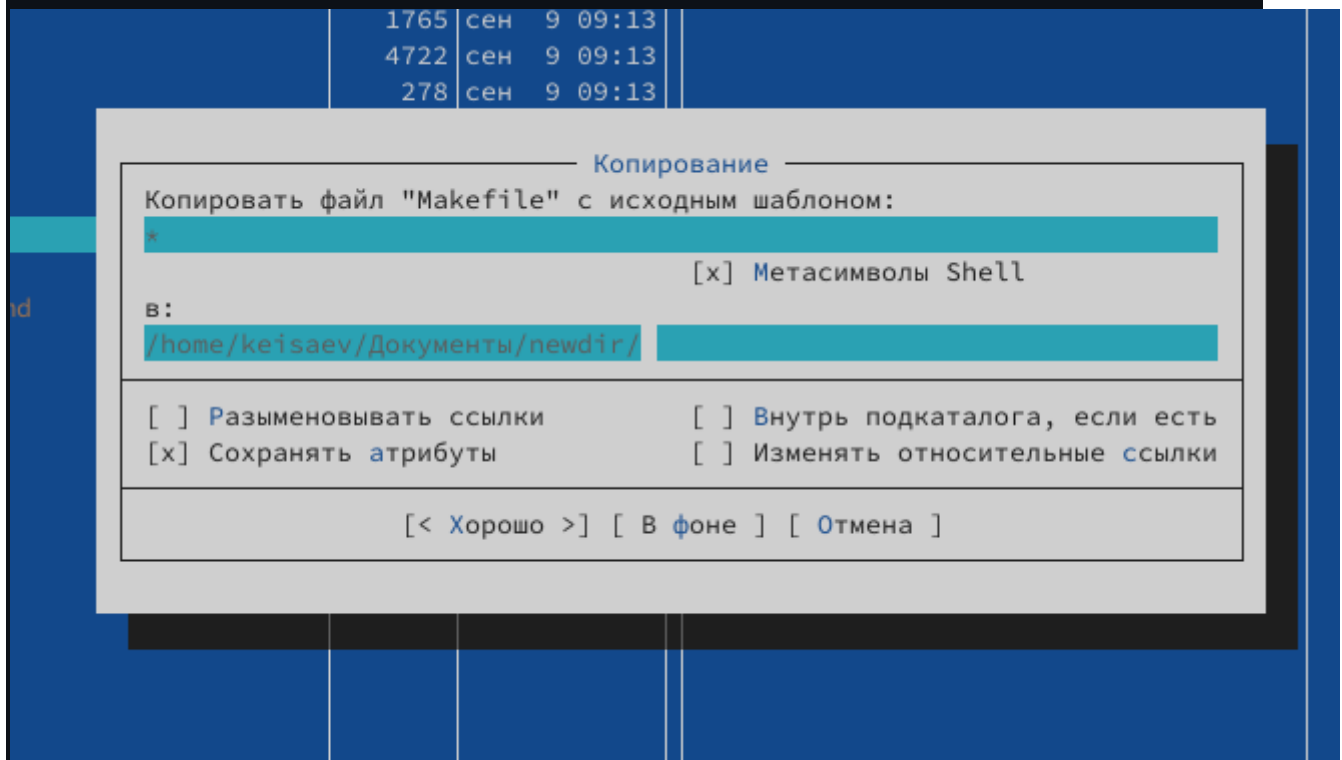
Работа с Midnight Commander



{ #fig:012

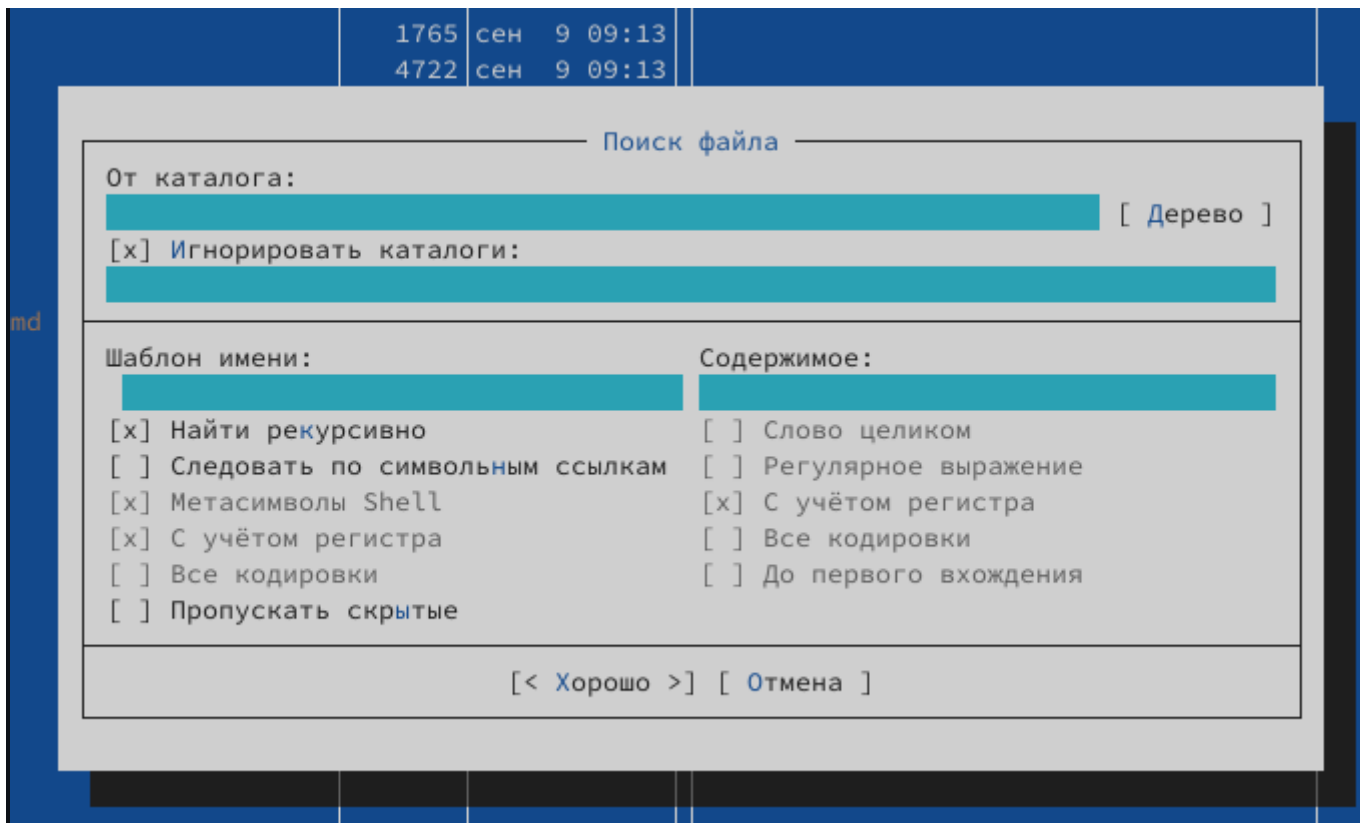
width=70% height=70% }

Работа с Midnight Commander



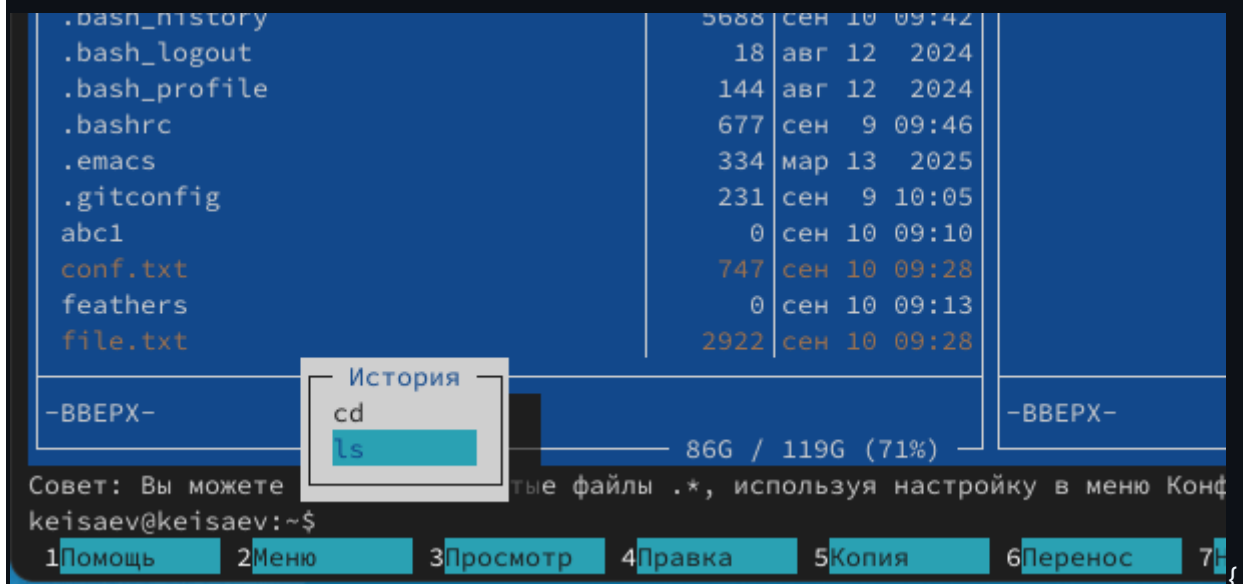
{ #fig:013 width=70% height=70% }

Работа с Midnight Commander



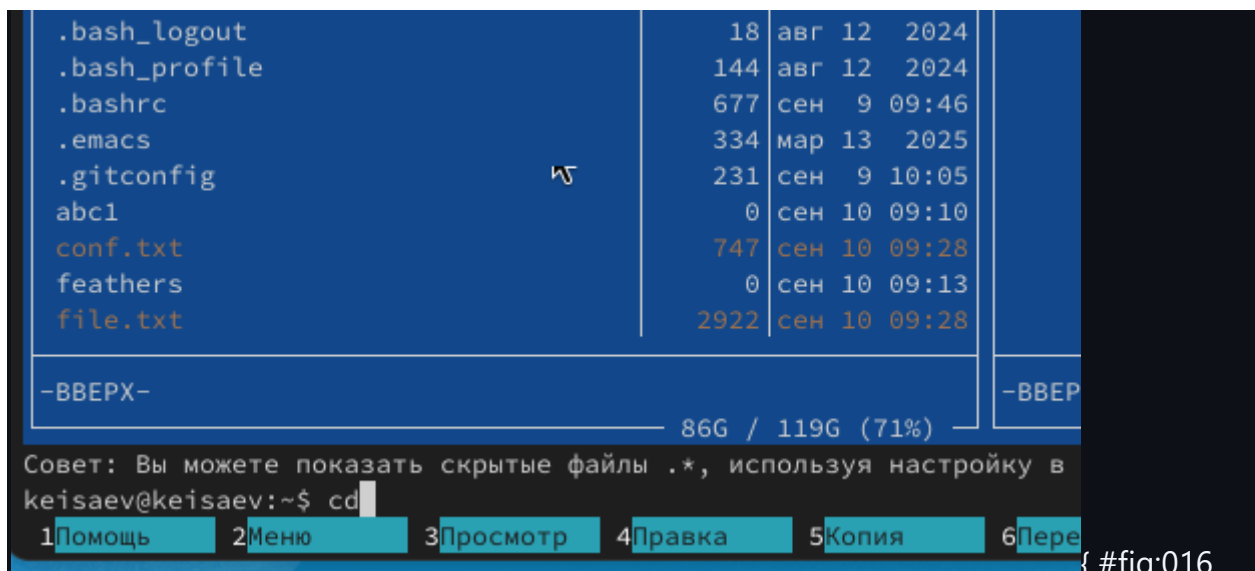
{ #fig:014 width=70% height=70% }

Работа с Midnight Commander



#fig:015 width=70% height=70% }

Работа с Midnight Commander



{ #fig:016

width=70% height=70% }

Работа с Midnight Commander

```
mc.ext.ini      [----]  0 L:[  1+ 0   1/1136] *(0   /26909b) 0035 0x023
# Midnight Commander 4.0 extension file
#
# Warning: The structure of this file has been completely changed with the version 4
#
# 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 (F9-c-e). If you edit this file in any other way
# while mc is running, you will need to press F9-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/xxxxx
# 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.
#
```

{ #fig:017 width=70% height=70% }

Работа с Midnight Commander


```

menu      [----]  1 L:[  1+16  17/370] *(732 /11821b) 0010 0x00A
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 Runs the commands and pipes standard output to the view command
#  If %view is immediately followed by '{', recognize keywords
#  ascii, hex, nroff and uniform
#
#  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
#  %lf     ditto
#  %0f     don't quote expanded macro
#####

+ ! t t
@      Do something on the current file
      CMD=%{Enter command}
      $CMD %f

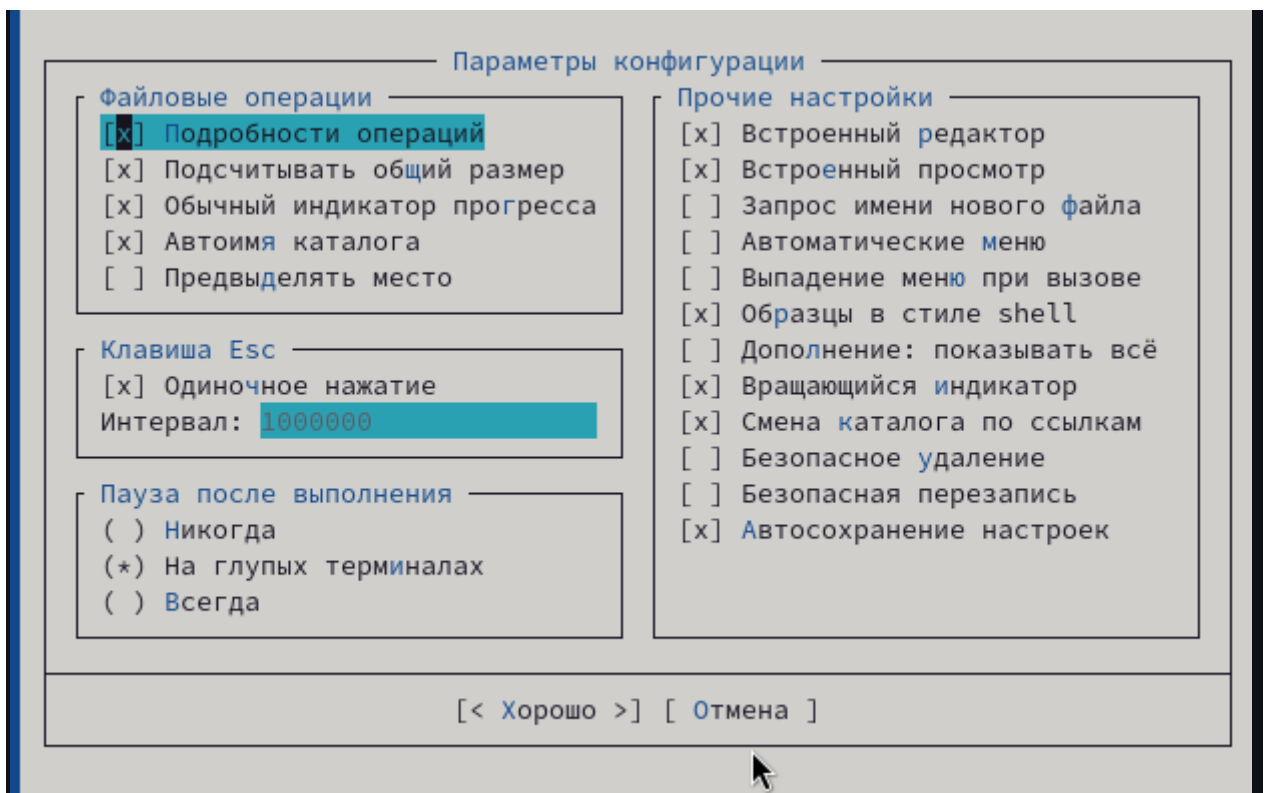
+ t t
@      Do something on the tagged files
      CMD=%{Enter command}
      for i in %t ; do

```

1Помощь 2Сохранить 3Блок 4Замена 5Копия 6Перем~тить 7Поиск

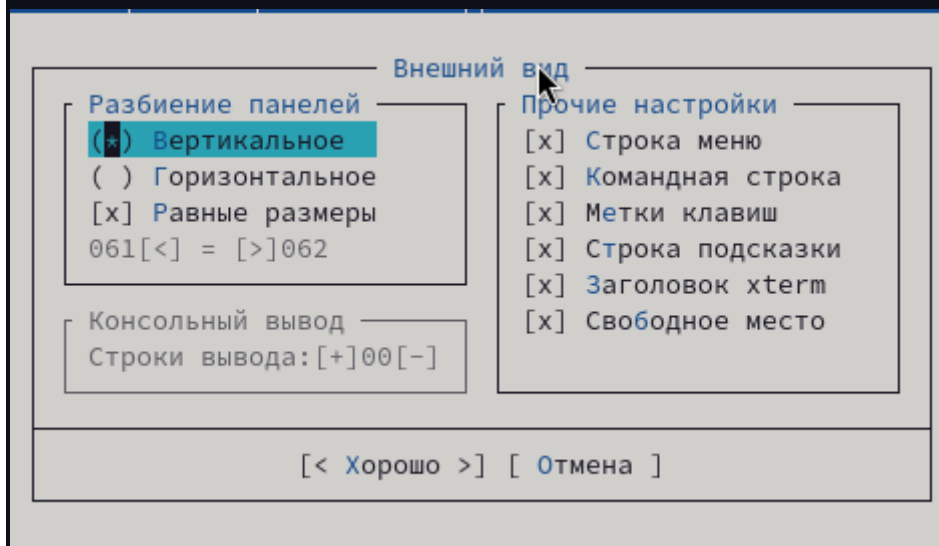
```
{ #fig:018 width=70% height=70% }
```

Работа с Midnight Commander



{ #fig:019 width=70% height=70% }

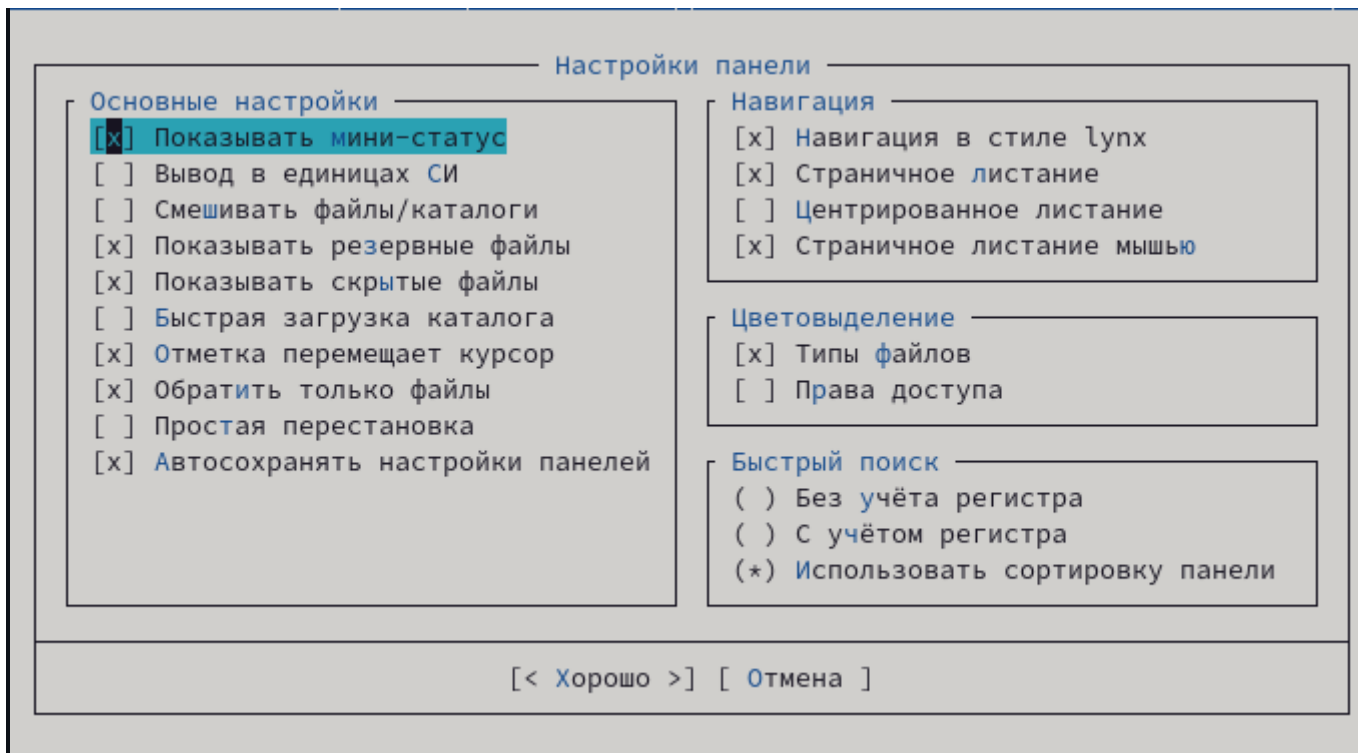
Работа с Midnight Commander



height=70% }

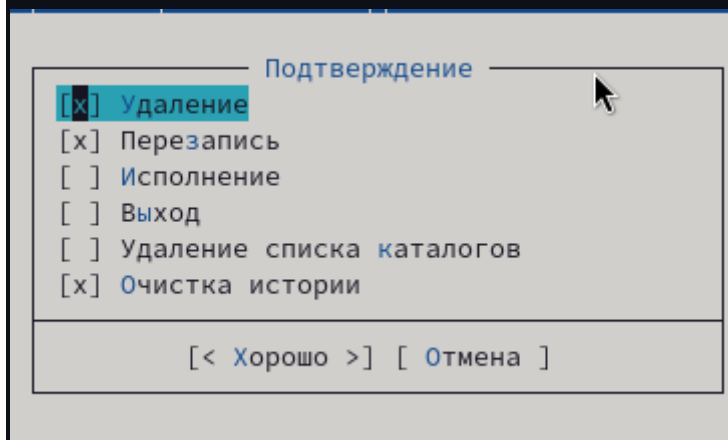
{ #fig:020 width=70%

Работа с Midnight Commander



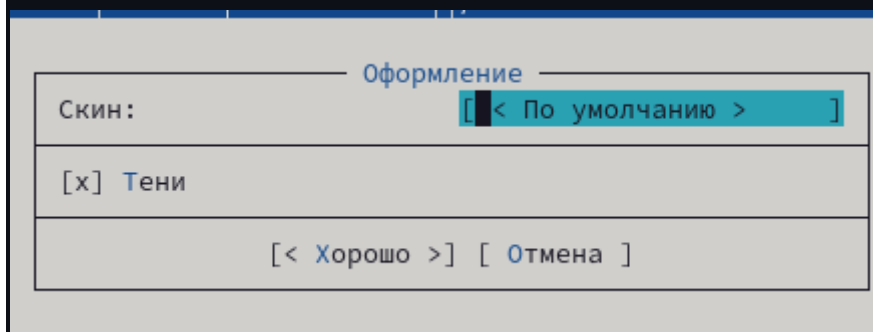
{ #fig:021 width=70% height=70% }

Работа с Midnight Commander



{ #fig:022 width=70% height=70% }

Работа с Midnight Commander



{ #fig:023 width=70%

height=70% }

Работа с Midnight Commander

Биты символов

Кодировка ввода/вывода: [UTF-8]
[x] Полный 8-битный ввод
[< Хорошо >] [Отмена]

{ #fig:024 width=70%

height=70% }

Работа с Midnight Commander

Определение клавиш

[Клавиша Esc]	[Функциональная 13]	[Стрелка влево]
[Функциональная 1]	[Функциональная 14]	[Стрелка вправо]
[Функциональная 2]	[Функциональная 15]	[Клавиша Insert]
[Функциональная 3]	[Функциональная 16]	[Клавиша Delete]
[Функциональная 4]	[Функциональная 17]	[Клавиша Home]
[Функциональная 5]	[Функциональная 18]	[Клавиша End]
[Функциональная 6]	[Функциональная 19]	[Клавиша Page Up]
[Функциональная 7]	[Функциональная 20]	[Клавиша Page Down]
[Функциональная 8]	[Завершение/M-tab]	[/ доп.клавиатуры]
[Функциональная 9]	[Обратная таб./S-tab]	[* доп.клавиатуры]
[Функциональная 10]	[Клавиша Backspace]	[- доп.клавиатуры]
[Функциональная 11]	[Стрелка вверх]	[+ доп.клавиатуры]
[Функциональная 12]	[Стрелка вниз]	

Нажмите все перечисленные выше клавиши. После нажатий найдите, какие клавиши не имеют пометки "OK". Для обучения клавише выберите её при помощи Tab или мышкой и нажмите пробел.

[< Сохранить >] [Отмена]

{ #fig:025 width=70% height=70% }

Работа с редактором Midnight Commander

```
mc [zasharofiddinov@zasharofiddinov]:~/Документы — /usr/bin/mc -P /var/tmp/mc-zashar
~/Документы
README.md [----] 0 L: [ 7+ 0 7/111] *(203 /4304b) 0035 0x023
# Основные идеи
- Стандартные соглашения об именах
- Стандартное соглашение для путей к файлам
- Стандартная настройка курса внутри шаблона курса

# Общие правила
- Рабочее пространство по предмету располагается в следующей иерархии:

  ``` bash
 ~/work/study/
 └─ <учебный год>/
 └─ <название предмета>/
 └─ <код предмета>/
  ```

- Например, для 2024--2025 учебного года и предмета «Операционные системы» (код пр

  ``` bash
 ~/work/study/
 └─ 2024-2025/
 └─ Операционные системы/
 └─ os-intro/
  ```

- Название проекта на хостинге git имеет вид:

  ``` example
 study_<учебный год>_<код предмета>
  ```

1Помощь 2Сохран 3Блок 4Замена 5Копия 6Пере~тить 7Поиск 8Уда
```

{ #fig:026 width=70% height=70% }

Работа с редактором Midnight Commander

```
mc [zasharofiddinov@zasharofiddinov]:~/Документы — /usr/bin/mc -P /var/tmp/mc-zashar
~/Документы
README.md [-M--] 0 L:[ 62+31 93/ 93] *(3829/3829b) <EOF>
## Сознание репозитория курса на основе шаблона

- Репозиторий на основе шаблона можно создать либо вручную, через web-интерфейс,
- Создание с помощью утилит выглядит следующим образом:

``` shell
gh repo create <new-repo-name> --template="<owner/template-repo>"
```

- Например, для 2024--2025 учебного года и предмета «Операционные системы» (код п

``` shell
mkdir -p ~/work/study/2024-2025/"Операционные системы"
cd ~/work/study/2024-2025/"Операционные системы"
gh repo create study_2024-2025_os-intro --template=yamadharm/course-directory-
git clone --recursive git@github.com:<owner>/study_2024-2025_os-intro.git os-in
```

``` shell
echo os-intro > COURSE
make
```

- Отправьте файлы на сервер:

``` shell
git add .
git commit -am 'feat(main): make course structure'
git push
```

1Помощь 2Сохран 3Блок 4Замена 5Копия 6Пере~тить 7Поиск 8Уда.
```

{ #fig:027 width=70% height=70% }

Работа с редактором Midnight Commander

```
mc [zasharofiddinov@zasharofiddinov]:~/Документы — /usr/bin/mc -P /var/tmp/mc-zashar
~/Документы
README.md [BM--] 0 L:[ 62+ 4 66/ 93] *(2797/3829b) 0045 0x02D
## Сознание репозитория курса на основе шаблона

- Репозиторий на основе шаблона можно создать либо вручную, через web-интерфейс,
- Создание с помощью утилит выглядит следующим образом:

''' shell
gh repo create <new-repo-name> --template="<owner/template-repo>"
'''

- Например, для 2024--2025 учебного года и предмета «Операционные системы» (код пр

''' shell
mkdir -p ~/work/study/2024-2025/"Операционные системы"
cd ~/work/study/2024-2025/"Операционные системы"
gh repo create study_2024-2025_os-intro --template=yamadharma/course-directory-s
git clone --recursive git@github.com:<owner>/study_2024-2025_os-intro.git os-int
'''

''' shell
echo os-intro > COURSE
make
'''

- Отправьте файлы на сервер:

''' shell
git add .
git commit -am 'feat(main): make course structure'
git push
'''

1Помощь 2Сохран 3Блок 4Замена 5Копия 6Пере~тить 7Поиск 8Удал
```

{ #fig:028 width=70% height=70% }

Работа с редактором Midnight Commander


```
mc [zasharofiddinov@zasharofiddinov]:~/Документы — /usr/bin/mc -P /var/tmp/mc-zashar
~/Документы
README.md [BM--] 0 L:[ 62+ 4 66/ 93] *(2797/3829b) 0045 0x02D
## Создание репозитория курса на основе шаблона

- Репозиторий на основе шаблона можно создать либо вручную, через web-интерфейс,
- Создание с помощью утилит выглядит следующим образом:

``` shell
gh repo create <new-repo-name> --template="<owner/template-repo>"
```

- Например, для 2024--2025 учебного года и предмета «Операционные системы» (код пр

``` shell
mkdir -p ~/work/stud
cd ~/work/stud
gh repo create
git clone --re
```

Сохранить файл
Подтверждаете запись файла "/home/zasharofiddinov/Документы/REA

[ Сохранить ] [ Отмена ]

``` shell
``` shell
echo os-intro > COURSE
make
```

- Отправьте файлы на сервер:

``` shell
git add .
git commit -am 'feat(main): make course structure'
git push
```

1Помощь 2Сохран 3Блок 4Замена 5Копия 6Пере~тить 7Поиск 8Удал
```

{ #fig:029 width=70% height=70% }

## Работа с редактором Midnight Commander

```
mc [zasharofiddinov@zasharofiddinov]:~/Документы — /usr/bin/mc -P /var/tmp/mc-zasharofiddinov
~/Документы

README.md [-M--] 0 L:[42+31 73/ 93] *(3242/3829b) 0010 0x00A
study_2024-2025_os-intro
...

- Каталог для лабораторных работ имеет вид `labs`.
- Каталоги для лабораторных работ имеют вид `lab<номер>`, например: `lab01`, `lab02`.
- Каталог для групповых проектов имеет вид `group-project`.
- Каталог для персональных проектов имеет вид `personal-project`.
- Если проектов несколько, то они нумеруются подобно лабораторным работам.
- Этапы проекта обозначаются как `stage<номер>`.

Шаблон для рабочего пространства

- Репозиторий:
 <https://github.com/yamadharm/course-directory-student-template>.

Сознание репозитория курса на основе шаблона

- Репозиторий на основе шаблона можно создать либо вручную, через web-интерфейс, либо с помощью утилит.
- Создание с помощью утилит выглядит следующим образом:

  ``` shell
  gh repo create <new-repo-name> --template="<owner/template-repo>"
  ```

- Например, для 2024--2025 учебного года и предмета «Операционные системы» (код предмета: 030) можно создать репозиторий с именем `2024-2025_030_операционные_системы`.
```

{ #fig:030 width=70% height=70% }

## Работа с редактором Midnight Commander

```
mc [zasharofiddinov@zasharofiddinov]:~/Документы — /usr/bin/mc -P /var/tmp/mc-zashar
~/Документы
README.md [-M--] 0 L: [62+31 93/ 93] *(3829/3829b) <EOF>
Сознание репозитория курса на основе шаблона

- Репозиторий на основе шаблона можно создать либо вручную, через web-интерфейс, л

- Создание с помощью утилит выглядит следующим образом:

    ``` shell
    gh repo create <new-repo-name> --template="<owner/template-repo>"
    ```

- Например, для 2024--2025 учебного года и предмета «Операционные системы» (код пр

    ``` shell
    mkdir -p ~/work/study/2024-2025/"Операционные системы"
    cd ~/work/study/2024-2025/"Операционные системы"
    gh repo create study_2024-2025_os-intro --template=yamadharma/course-directory-s
    git clone --recursive git@github.com:<owner>/study_2024-2025_os-intro.git os-int
    ```

    ``` shell
    echo os-intro > COURSE
    make
    ```

- Отправьте файлы на сервер:

    ``` shell
    git add .
    git commit -am 'feat(main): make course structure'
    git push
    ```

1Помощь 2Сохран 3Блок 4Замена 5Копия 6Пере~тить 7Поиск 8Удал
```

{ #fig:031 width=70% height=70% }

## Работа с редактором Midnight Commander

```
mc [zasharofiddinov@zasharofiddinov]:~/Документы — /usr/bin/mc -P /var/tmp/mc-zasharofiddinov
~/Документы
README.md [-M--] 35 L: [1+ 0 1/ 93] *(64 /3829b) 1086 0x43E
Шаблон каталога курса для студентов

Скачать репозиторий

git clone --recursive https://github.com/yamadharma/course-directory-student-template

Основные идеи

- Стандартные соглашения об именах
- Стандартное соглашение для путей к файлам
- Стандартная настройка курса внутри шаблона курса

Общие правила

- Рабочее пространство по предмету располагается в следующей иерархии:

    ``` bash
    ~/work/study/
    └─ <учебный год>/
       └─ <название предмета>/
          └─ <код предмета>/
    ```

- Например, для 2024--2025 учебного года и предмета «Операционные системы» (код пр

    ``` bash
    ~/work/study/
    └─ 2024-2025/
       └─ Операционные системы/
          └─ os-intro/
    ```

1Помощь 2Сохран 3Блок 4Замена 5Копия 6Пере~тить 7Поиск 8Удал
```

{ #fig:032 width=70% height=70% }

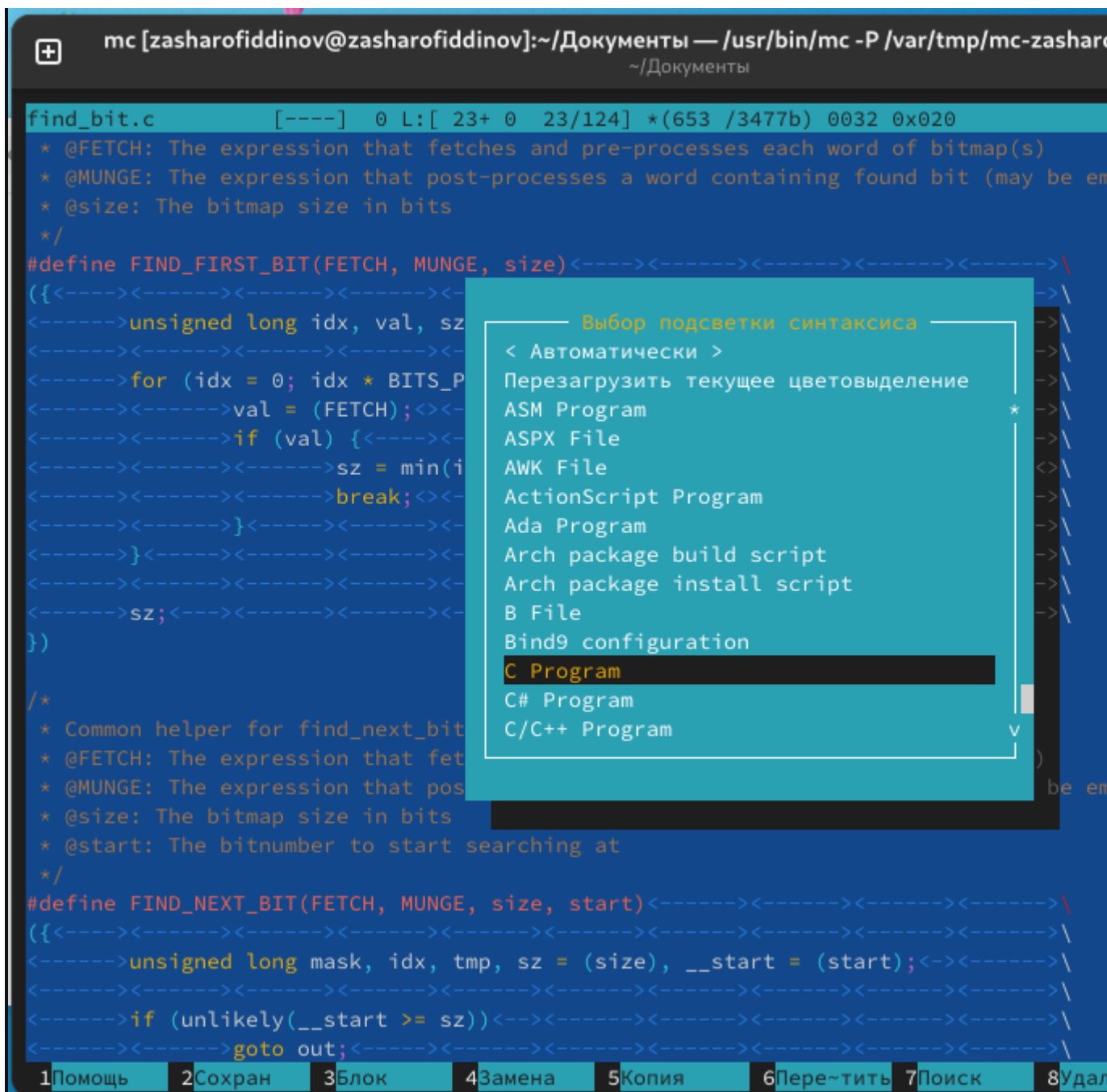
## Работа с редактором Midnight Commander

```
mc [zasharofiddinov@zasharofiddinov]:~/Документы — /usr/bin/mc -P /var/tmp/mc-zashar
~/Документы
find_bit.c [----] 0 L:[23+ 0 23/124] *(653 /3477b) 0032 0x020
/* @FETCH: The expression that fetches and pre-processes each word of bitmap(s)
 * @MUNGE: The expression that post-processes a word containing found bit (may be e
 * @size: The bitmap size in bits
 */
#define FIND_FIRST_BIT(FETCH, MUNGE, size)<-----><-----><-----><-----><----->\
({<-----><-----><-----><-----><-----><-----><-----><-----><-----><----->\
<----->unsigned long idx, val, sz = (size);<--><-----><-----><-----><----->\
<-----><-----><-----><-----><-----><-----><-----><-----><-----><----->\
<----->for (idx = 0; idx * BITS_PER_LONG < sz; idx++) {<-----><-----><----->\
<-----><----->val = (FETCH);<-----><-----><-----><-----><-----><-----><----->\
<-----><----->if (val) {<-----><-----><-----><-----><-----><-----><----->\
<-----><-----><----->sz = min(idx * BITS_PER_LONG + __ffs(MUNGE(val)), sz);<-->\
<-----><-----><----->break;<-----><-----><-----><-----><-----><-----><----->\
<-----><----->}<-----><-----><-----><-----><-----><-----><-----><----->\
<----->}<-----><-----><-----><-----><-----><-----><-----><-----><----->\
<-----><-----><-----><-----><-----><-----><-----><-----><-----><----->\
<----->sz;<-----><-----><-----><-----><-----><-----><-----><-----><----->\
})

/*
 * Common helper for find_next_bit() function family
 * @FETCH: The expression that fetches and pre-processes each word of bitmap(s)
 * @MUNGE: The expression that post-processes a word containing found bit (may be e
 * @size: The bitmap size in bits
 * @start: The bitnumber to start searching at
 */
#define FIND_NEXT_BIT(FETCH, MUNGE, size, start)<-----><-----><-----><-----><----->\
({<-----><-----><-----><-----><-----><-----><-----><-----><-----><----->\
<----->unsigned long mask, idx, tmp, sz = (size), __start = (start);<--><----->\
<-----><-----><-----><-----><-----><-----><-----><-----><-----><----->\
<----->if (unlikely(__start >= sz))<--><-----><-----><-----><-----><-----><----->\
<-----><----->goto out;<-----><-----><-----><-----><-----><-----><-----><----->\
1Помощь 2Сохран 3Блок 4Замена 5Копия 6Пере~тить 7Поиск 8Уда
```

{ #fig:033 width=70% height=70% }

## Работа с редактором Midnight Commander



{ #fig:034 width=70% height=70% }

## Выводы по проделанной работе

### Вывод

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