

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра фундаментальной информатики и информационных технологий

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №5

Тема: Основы работы с Midnight Commander

Дисциплина: Архитектура компьютеров

Студент: Герчет Вячеслав

Группа: НКАбд-03-25

Студ. билет № 1132255650

Преподаватель: Штепа Кристина Александровна

МОСКВА

2025 г.

Содержание

1. Цель работы
2. Задание
3. Теоретическое введение
4. Выполнение лабораторной работы
 - 4.1 Работа с Midnight Commander
 - 4.2 Создание программы lab5-1.asm
 - 4.3 Создание программы lab5-2.asm
5. Выполнение самостоятельной работы
6. Выводы
7. Список литературы

Список иллюстраций

- 3.1 Вводим в консоль команду `mc`
- 3.2 Переходим в каталог
- 3.3 Создаем каталог функциональной клавишей F7
- 3.4 Воспользуемся командой `touch`
- 3.5 Открываем файл функциональной клавишей, заполняем и сохраняем
- 3.6 Открываем файл и убеждаемся, что файл содержит текст программы
- 3.7 Проверяем, как работает данная программа
- 3.8 Скачиваем файл
- 3.9 Копируем скаченный файл
- 3.10 Создаем копию файла клавишей F6
- 3.11 Проверяем скопировался ли файл
- 3.12 Открываем и заполняем файл
- 3.13 Смотрим, как сработала программа
- 3.14 Редактируем файл
- 3.15 Смотрим, как сработал программа и сравниваем с прошлой
- 3.16 Создаем копию файла `lab5-1.asm`
- 3.17 Редактируем файл
- 3.18 Проверяем правильность написания программы

1. Цель работы

Изучить основы работы с файловым менеджером `Midnight Commander`, закрепить навыки написания программ на языке ассемблера `NASM` и ознакомиться с системными вызовами в ОС `GNU/Linux`.

2. Задание

Написать 2 программы по примеру и впоследствии изменить их по условию.

3. Теоретическое введение

Midnight Commander (mc) — это текстовый файловый менеджер, предназначенный для удобной навигации по файловой системе Linux. Он позволяет создавать, копировать, редактировать и удалять файлы с помощью функциональных клавиш (F1–F10). Работа с ним ведётся в двухпанельном интерфейсе.

NASM (Netwide Assembler) — популярный ассемблер с синтаксисом Intel. Программы на NASM обычно содержат три секции:

data — хранит инициализированные данные;
bss — хранит неинициализированные данные;
text — содержит исполняемый код программы.

Для обращения к системным функциям Linux используются прерывания int 80h. Наиболее часто применяемые вызовы:

4 — sys_write (вывод);
3 — sys_read (ввод);
1 — sys_exit (завершение программы).

4. Выполнение лабораторной работы

4.1 Работа с Midnight Commander

Открываем Midnight Commander (рис. 3.1).

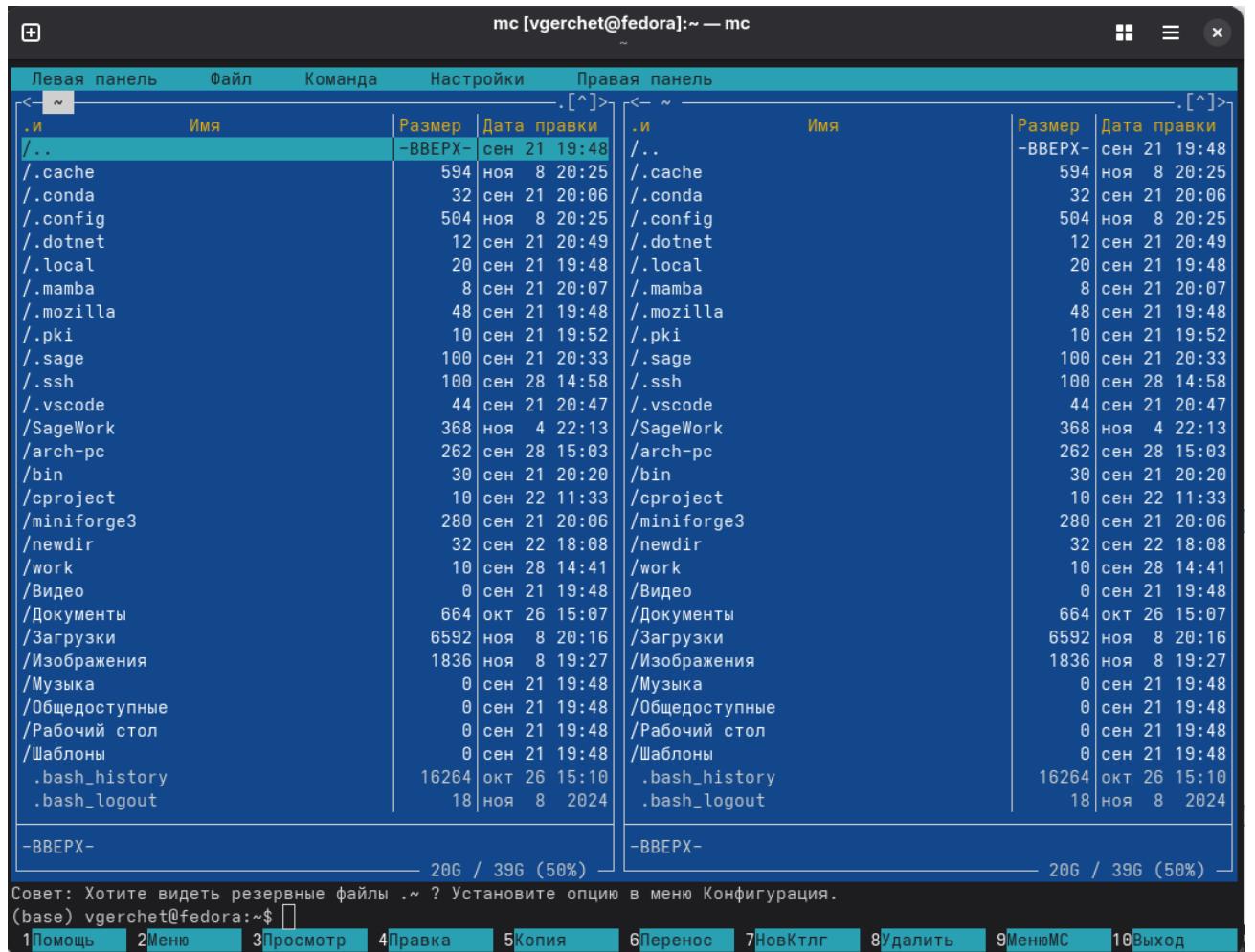


Рис. 3.1: Вводим в консоль команду mc

Переходим в каталог, созданный при выполнении 4 ЛБ (рис. 3.2).

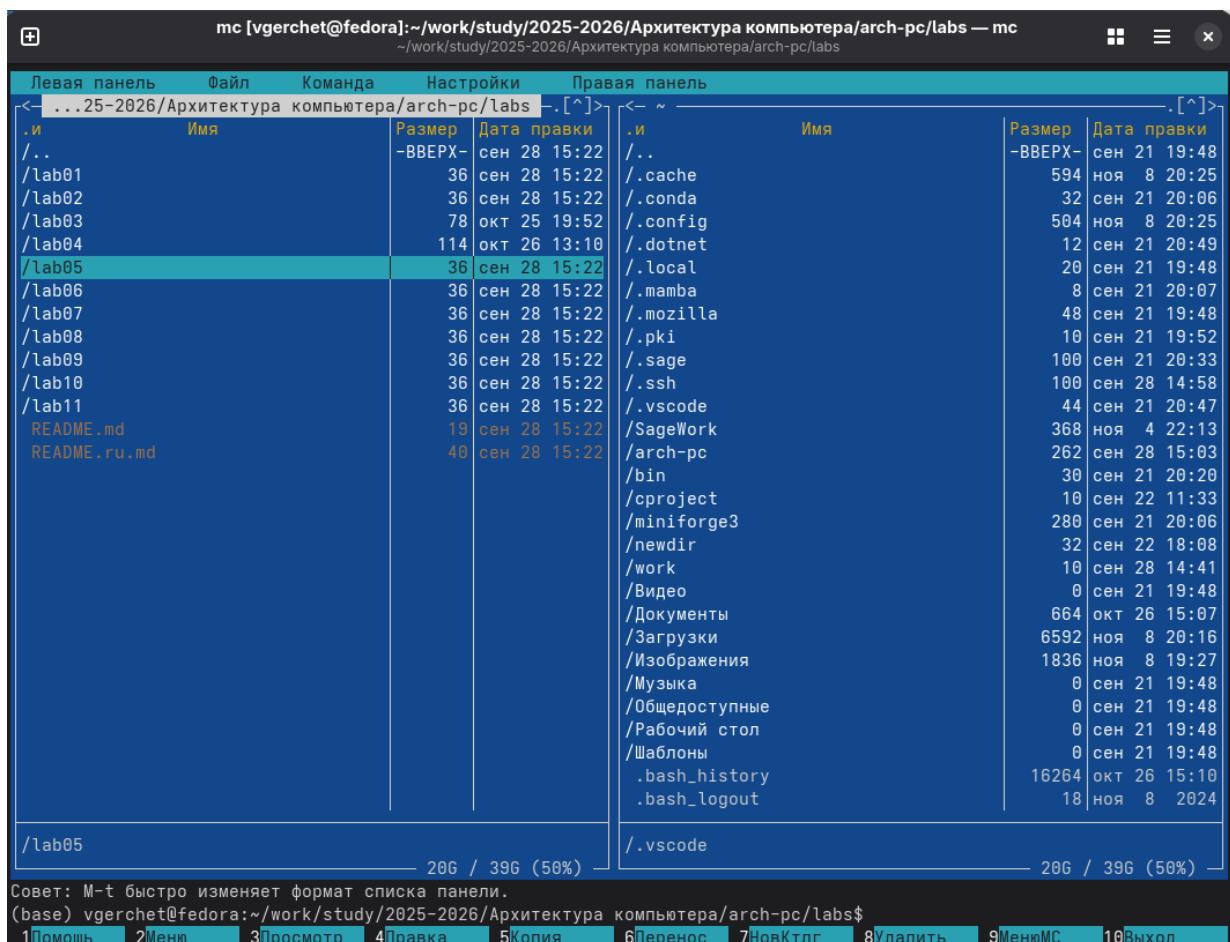


Рис. 3.2: Переходим в каталог

Создаем каталог lab05 (рис. 3.3).

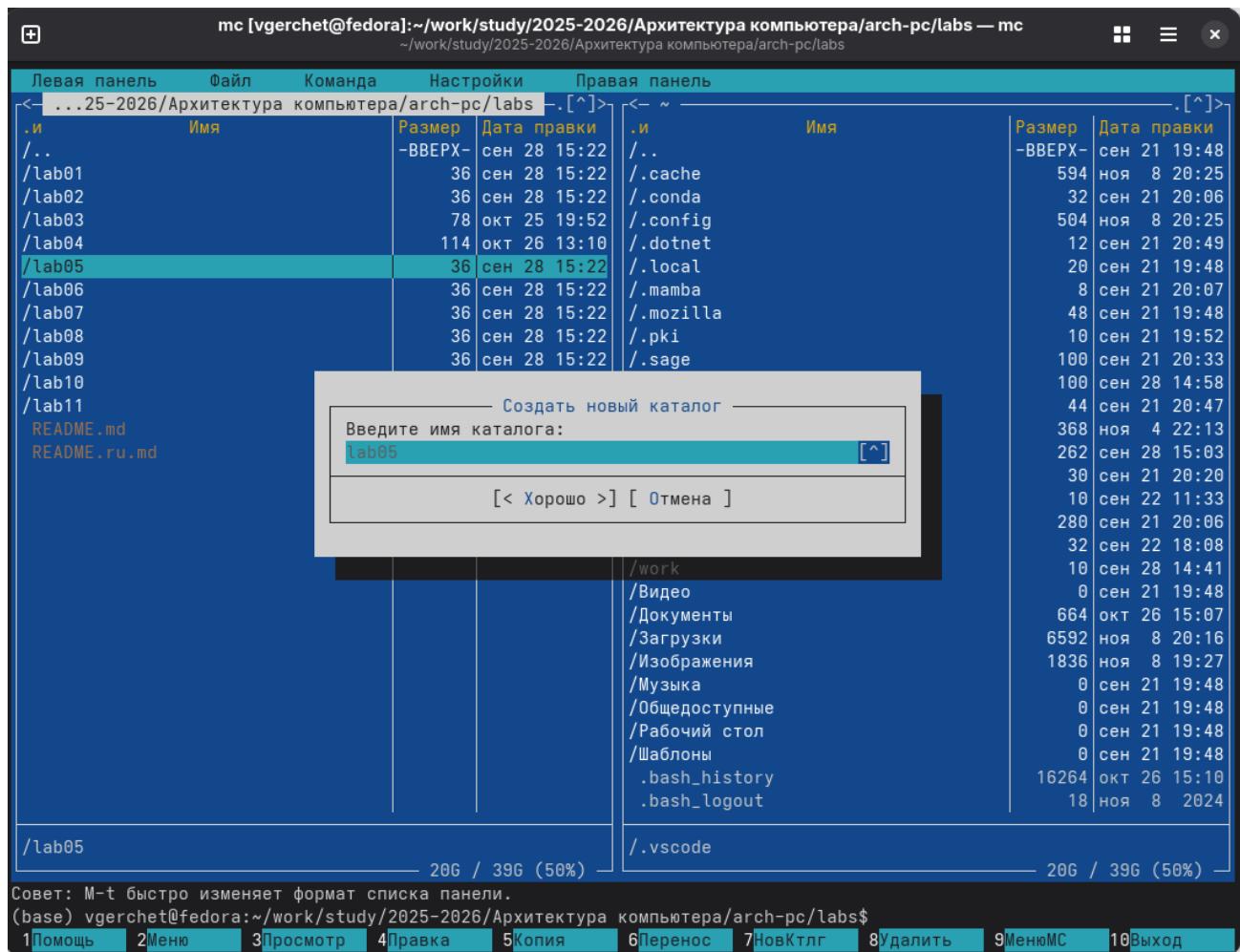


Рис. 3.3: Создаем каталог функциональной клавишей F7

4.2 Создание программы lab5-1.asm

Создаем файл lab5-1.asm (рис. 3.4).

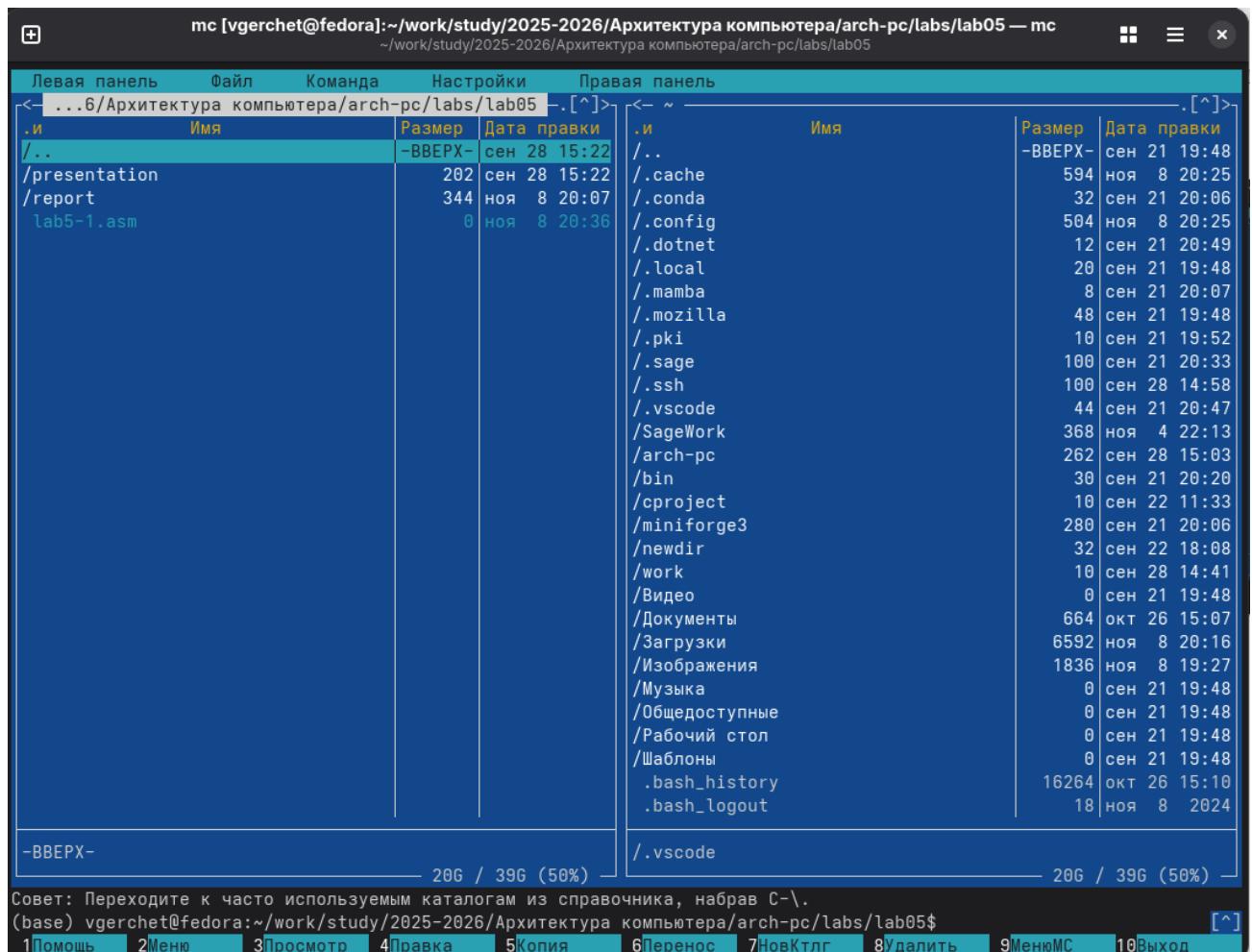


Рис. 3.4: Воспользуемся командой touch

Открываем файл для редактирования и заполняем его по листингу (рис. 3.5).

The screenshot shows a terminal window titled "mc [vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05 — mc". The file being edited is "lab5-1.asm". The assembly code is as follows:

```
GNU nano 8.3 /home/vgerchet/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05 — mc
~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05

; lab5-1.asm — программа ввода и вывода строки
; Автор: Герчет Вячеслав, НКАбд-03-25
; Лабораторная №5, Архитектура компьютеров

SECTION .data
msg: DB 'Введите строку: ', 0Ah      ; сообщение приглашения
msgLen: EQU $ - msg

SECTION .bss
buf: RESB 80                      ; буфер для ввода строки

SECTION .text
GLOBAL _start

_start:
; Вывод приглашения
mov eax, 4            ; системный вызов write
mov ebx, 1            ; дескриптор stdout
mov ecx, msg          ; адрес сообщения
mov edx, msgLen       ; длина сообщения
int 80h

; Чтение строки
mov eax, 3            ; системный вызов read
mov ebx, 0            ; stdin
mov ecx, buf          ; адрес буфера
mov edx, 80           ; размер буфера
int 80h

; Вывод введённой строки
mov eax, 4
mov ebx, 1
mov ecx, buf
mov edx, 80

[Прочитана 41 строка]

[Keyboard Shortcuts]
^G Справка      ^O Записать      ^F Поиск      ^K Вырезать      ^T Выполнить      ^C Позиция      M-U Отмена
^X Выход      ^R ЧитФайл      ^\ Замена      ^U Вставить      ^J Выровнять      ^/ К строке      M-E Повтор
```

Рис. 3.5: Открываем файл функциональной клавишей, заполняем и сохраняем

Открываем файл для просмотра (рис. 3.6).

```
vgerchet@fedora:~  
~  
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05$ cat lab5-1.asm  
; lab5-1.asm – программа ввода и вывода строки  
; Автор: Герчет Вячеслав, НКАбд-03-25  
; Лабораторная №5, Архитектура компьютеров  
  
SECTION .data  
msg: DB 'Введите строку: ', 0Ah      ; сообщение приглашения  
msgLen: EQU $ - msg  
  
SECTION .bss  
buf: RESB 80                      ; буфер для ввода строки  
  
SECTION .text  
GLOBAL _start  
  
.start:  
    ; Вывод приглашения  
    mov eax, 4          ; системный вызов write  
    mov ebx, 1          ; дескриптор stdout  
    mov ecx, msg        ; адрес сообщения  
    mov edx, msgLen    ; длина сообщения  
    int 80h  
  
    ; Чтение строки  
    mov eax, 3          ; системный вызов read  
    mov ebx, 0          ; stdin  
    mov ecx, buf        ; адрес буфера  
    mov edx, 80          ; размер буфера  
    int 80h  
  
    ; Вывод введённой строки  
    mov eax, 4          ; системный вызов write  
    mov ebx, 1          ; дескриптор stdout  
    mov ecx, buf        ; адрес буфера  
    mov edx, 80          ; размер буфера  
    int 80h  
  
    ; Завершение программы
```

Рис. 3.6: Открываем файл и убеждаемся, что файл содержит текст программы

Транслируем текст программы и запускаем исполняемый файл (рис. 3.7).

```
vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05
~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05

(base) vgerchet@fedora:~$ ~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05
bash: /home/vgerchet/work/study/2025-2026/Архитектура: Нет такого файла или каталога
(base) vgerchet@fedora:~$ cd ~/work/study/2025-2026/Архитектура\ компьютера/arch-pc/labs/lab05
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура\ компьютера/arch-pc/labs/lab05$ ls
lab5-1.asm presentation report
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура\ компьютера/arch-pc/labs/lab05$ nasm -f elf lab5-1.asm
ld -m elf_i386 lab5-1.o -o lab5-1
./lab5-1
Введите строку:
Герчет Вячеслав
Герчет Вячеслав
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура\ компьютера/arch-pc/labs/lab05$
```

Рис. 3.7: Проверяем, как работает данная программа

4.3 Создание программы lab5-2.asm

Скачиваем файл со страницы курса (рис. 3.8).

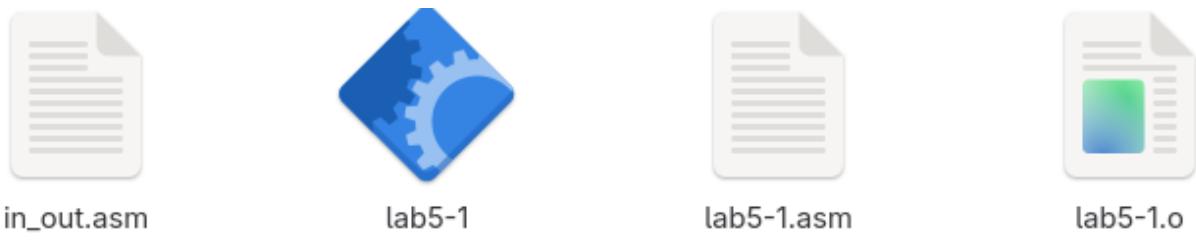
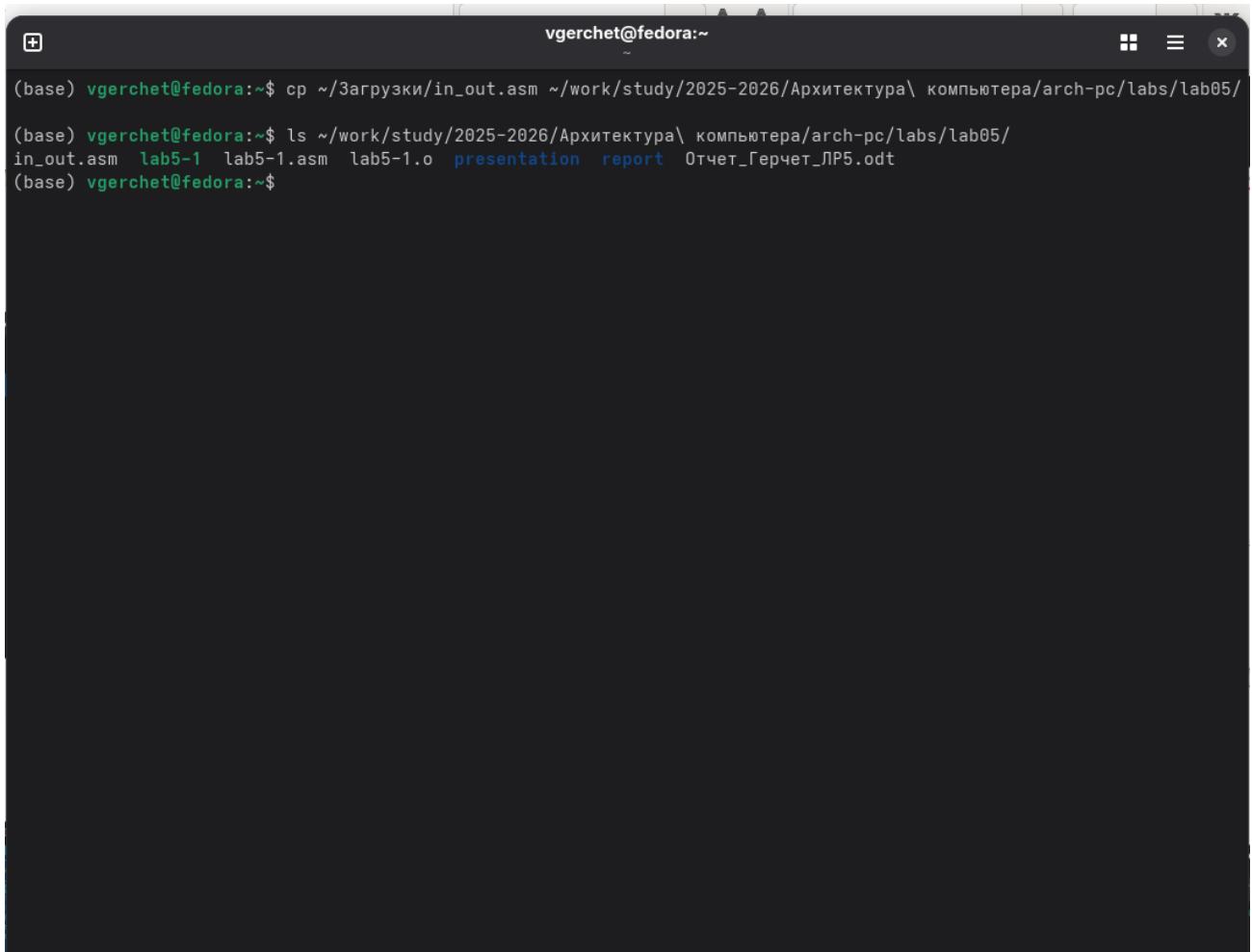


Рис. 3.8: Скачиваем файл

Копируем файл в нужную директорию (рис. 3.9).



```
vgerchet@fedora:~  
(base) vgerchet@fedora:~$ cp ~/Загрузки/in_out.asm ~/work/study/2025-2026/Архитектура\ компьютера/arch-pc/labs/lab05/  
(base) vgerchet@fedora:~$ ls ~/work/study/2025-2026/Архитектура\ компьютера/arch-pc/labs/lab05/  
in_out.asm lab5-1 lab5-1.asm lab5-1.o presentation report Отчет_Герчет_ЛР5.odt  
(base) vgerchet@fedora:~$
```

Рис. 3.9: Копируем скаченный файл

Создаем копию файла lab5-1.asm (рис. 3.10).

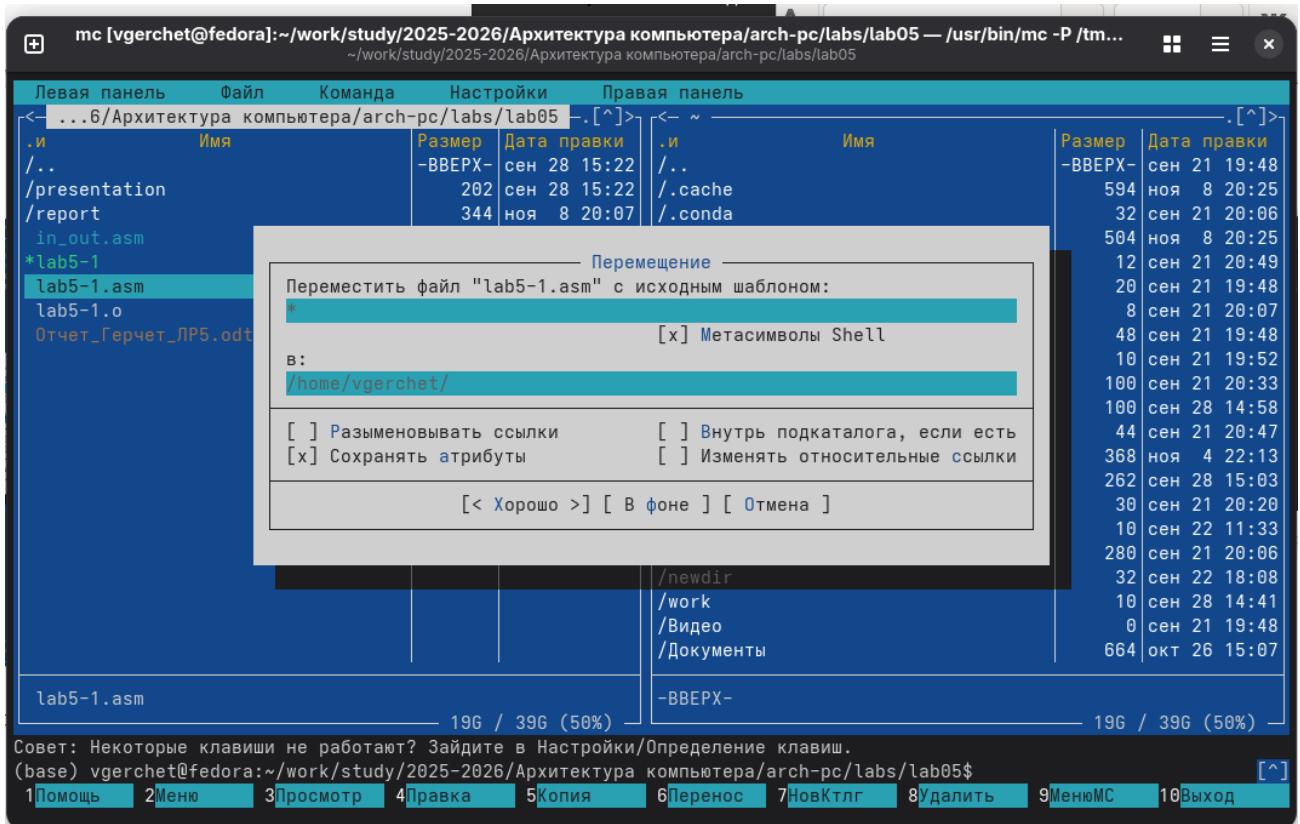


Рис. 3.10: Создаем копию файла клавишей F6

Проверяем созданный файл (рис. 3.11).

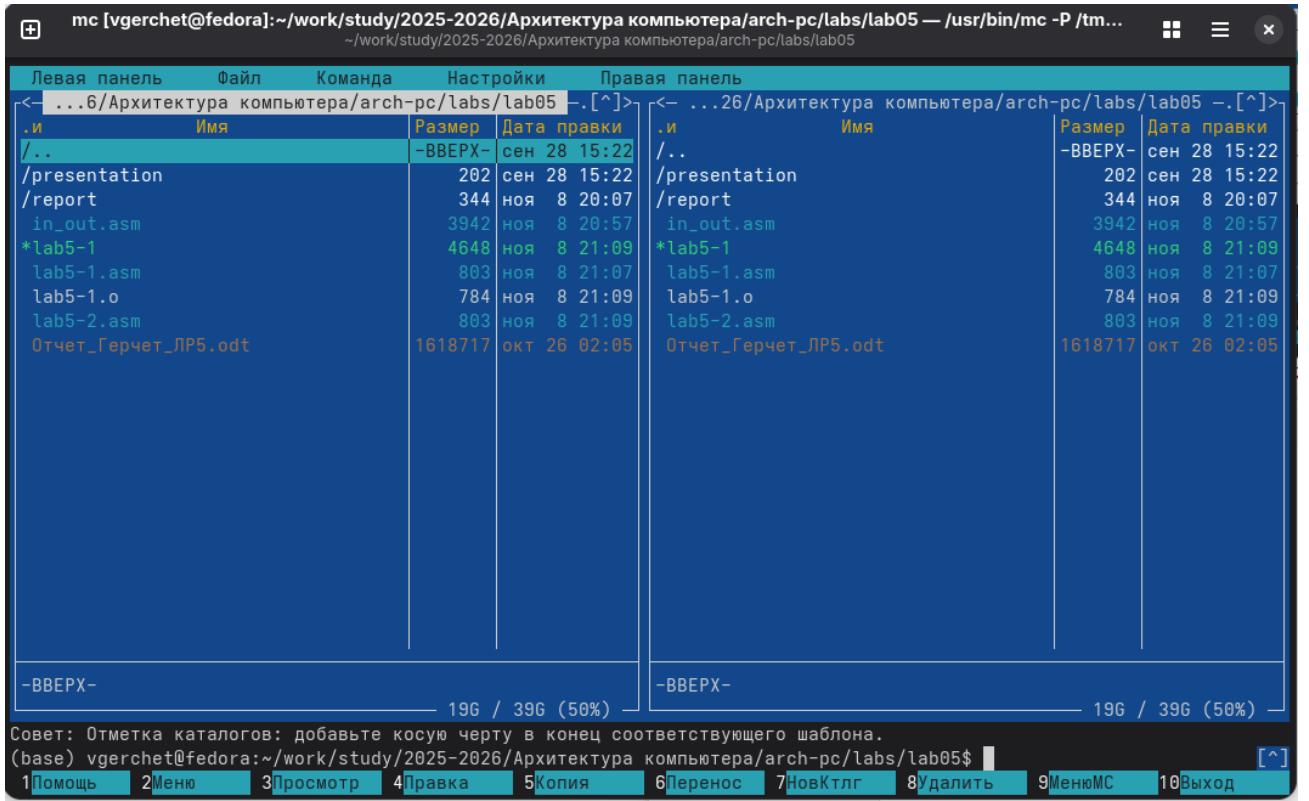


Рис. 3.11: Проверяем скопировался ли файл

Открываем новый файл и заполняем его в соответствии с листингом (рис. 3.12).

The screenshot shows a terminal window titled 'mc [vgerchet@fedora]'. The command entered is 'mc /home/vgerchet/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05 — /usr/bin/mc -P /tm... ~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05'. The file being edited is 'GNU nano 8.3 /home/vgerchet/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05/lab5-2.asm'. The assembly code in the editor is:

```
GNU nano 8.3 /home/vgerchet/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05 — /usr/bin/mc -P /tm...
~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05/lab5-2.asm

%include 'in_out.asm'
SECTION .data
msg: DB 'Введите строку: ',0h
SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:
    mov eax, msg
    call sprintLF
    mov ecx, buf1
    mov edx, 80
    call sread
    mov eax, buf1
    call sprint
    call quit
```

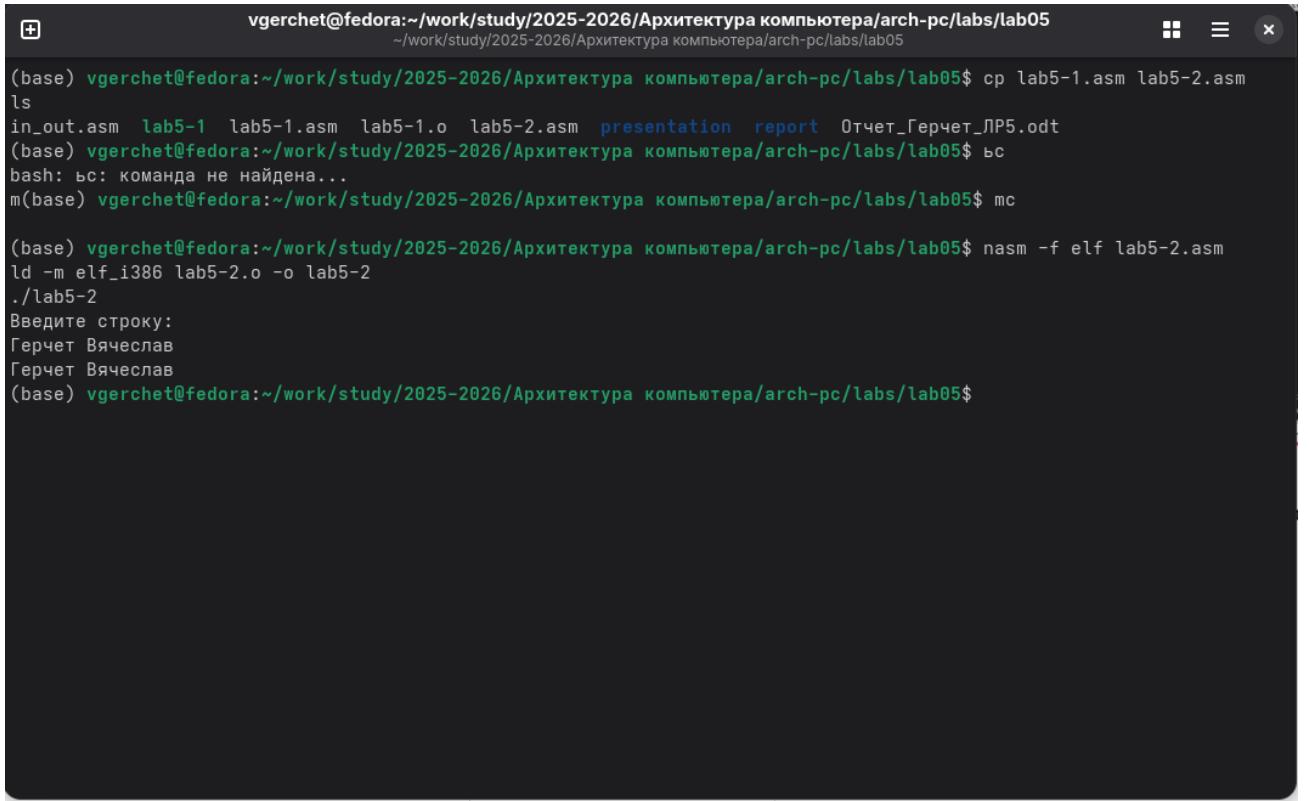
At the bottom of the window, there is a menu bar with Russian labels and keyboard shortcuts:

- Справка (Ctrl+G)
- Записать (Ctrl+O)
- Поиск (Ctrl+F)
- Вырезать (Ctrl+K)
- Выполнить (Ctrl+T)
- Позиция (Ctrl+C)
- Отмена (M-U)
- Выход (Ctrl+X)
- читФайл (Ctrl+R)
- Замена (Ctrl+\)
- Вставить (Ctrl+U)
- Выровнять (Ctrl+J)
- К строке (Ctrl+/)
- Повтор (M-E)

A status bar at the bottom right indicates '[Прочитано 17 строк]'

Рис. 3.12: Открываем и заполняем файл

Транслируем и запускаем новый файл (рис. 3.13).



```
vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05
~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05

(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05$ cp lab5-1.asm lab5-2.asm
ls
in_out.asm  lab5-1.asm  lab5-1.o  lab5-2.asm  presentation_report  Отчет_Герчет_ЛР5.odt
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05$ ьс
bash: ьс: команда не найдена...
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05$ mc

(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05$ nasm -f elf lab5-2.asm
ld -m elf_i386 lab5-2.o -o lab5-2
./lab5-2
Введите строку:
Герчет Вячеслав
Герчет Вячеслав
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05$
```

Рис. 3.13: Смотрим, как сработала программа

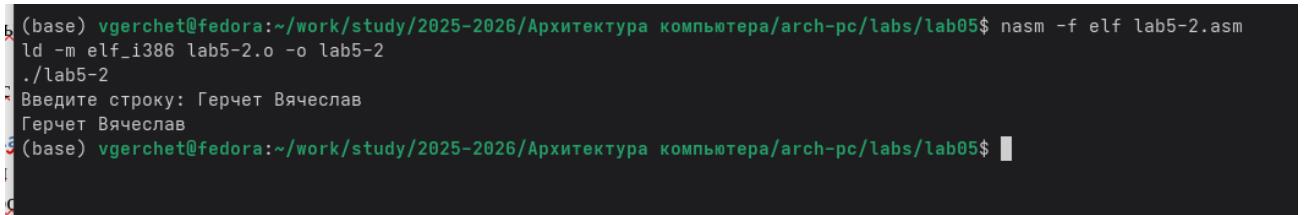
Снова открываем файл для редактирования и меняем sprintLF на sprint(рис. 3.14).



```
mov eax, msg
call sprint
mov ecx, buf1
mov edx, 80
call sread
mov eax, buf1
call sprint
call quit
```

Рис. 3.14: Редактируем файл

Транслируем и запускаем файл(рис. 3.15).



```
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05$ nasm -f elf lab5-2.asm
ld -m elf_i386 lab5-2.o -o lab5-2
./lab5-2
Введите строку: Герчет Вячеслав
Герчет Вячеслав
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05$
```

Рис. 3.15: Смотрим, как сработал программа и сравниваем с прошлой

Таким образом можем понять, что команда `sprint` выводит текст в той же строке, а `sprintLF` переносит на новую строку.

5. Выполнение самостоятельной работы

Создаем копию файла `lab5-1.asm` и называем его так же (рис. 3.16).

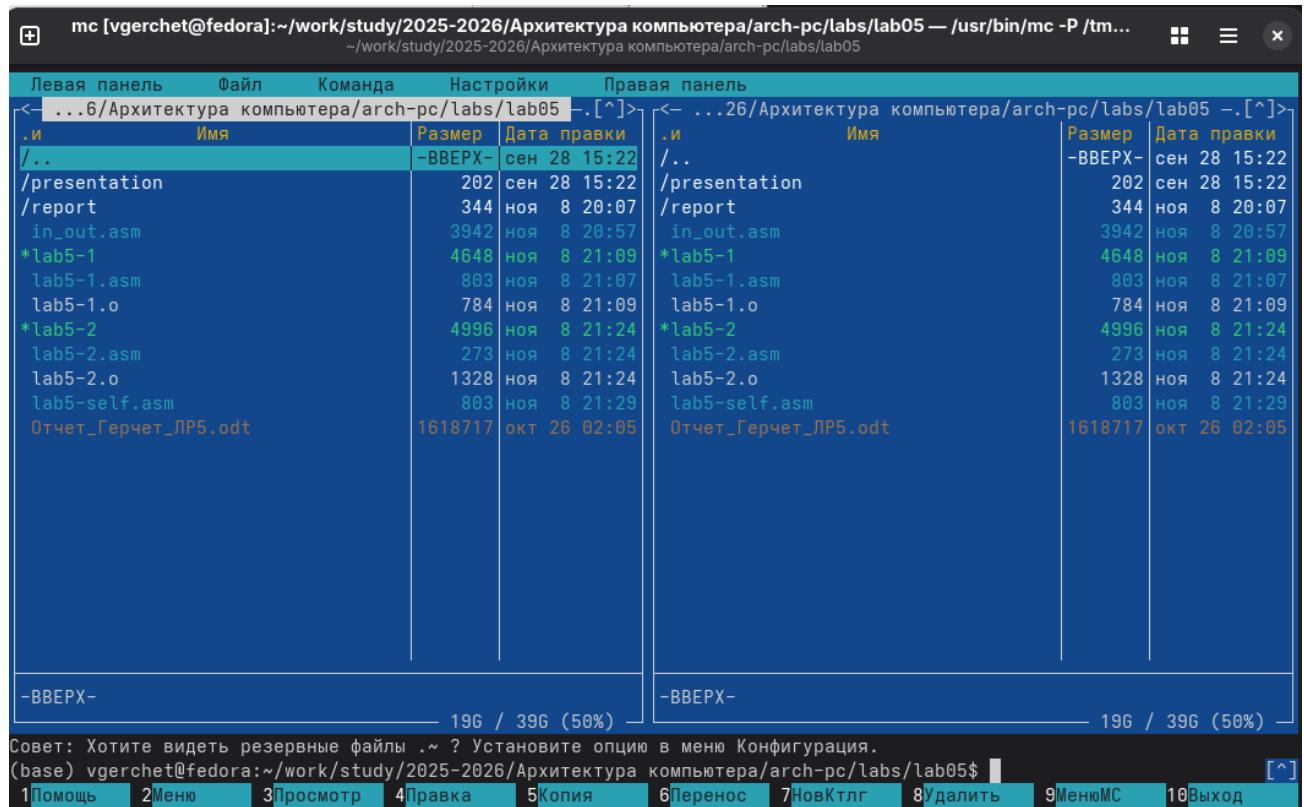


Рис. 3.16: Создаем копию файла `lab5-1.asm`

Редактируем файл, чтобы введенный текст с клавиатуры выводился в консоль (рис. 3.17).

The screenshot shows a terminal window titled 'mc [vgerchet@fedora]'. The command entered is 'mc vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05 — /usr/bin/mc -P /tm... ~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05'. The file being edited is 'GNU nano 8.3 /home/vgerchet/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05/lab5-self.asm'. The assembly code is as follows:

```
%include "in_out.asm"

SECTION .data
ask:    db 'Введите строку: ', 0
echo:   db 'Вы ввели: ', 0

SECTION .bss
buf:    resb 80

SECTION .text
GLOBAL _start

_start:
; Выводим приглашение
mov eax, ask
call sprintLF

; Считываем строку в буфер
mov ecx, buf
mov edx, 80
call sread

; Выводим фразу "Вы ввели: "
mov eax, echo
call sprint
```

At the bottom of the terminal, there is a menu bar with various keyboard shortcuts for file operations like 'Справка' (Help), 'Записать' (Save), 'Поиск' (Search), etc.

Рис. 3.17: Редактируем файл

Транслируем файл и запускаем программу (рис. 3.18).

The screenshot shows a terminal window with the following commands and output:

```
(base) vgerchet@fedora:~/work/study/2025-2026/Архитектура компьютера/arch-pc/labs/lab05$ nasm -f elf lab5-self.asm
ld -m elf_i386 lab5-self.o -o lab5-self
./lab5-self
Введите строку:
Герчет Вячеслав
Вы ввели: Герчет Вячеслав
```

Рис. 3.18: Проверяем правильность написания программы

4. Выводы

В результате выполнения лабораторной работы были освоены основы работы в Midnight Commander, а также принципы написания и компоновки программ на языке ассемблера NASM. Изучены системные вызовы Linux и подпрограммы ввода/вывода. Получены практические навыки работы с файлами и структурой программ на низком уровне.

5. Список литературы

1. NASM Assembly Language Tutorials — <https://cs.lmu.edu/~ray/notes/nasmtutorial/>
2. Midnight Commander Development Center - <https://midnight-commander.org/>
3. GNU Bash Manual — <https://www.gnu.org/s/bash/manual/bash.html>