

labrotary work Nº5

computer architecture

Erfan hosseinabadi

Content

1	Objective of the Work	5
2	Task	6
3	Theoretical Introduction	7
3.1	Conducting the Laboratory Work	8
3.1.1	Basics of Working with Midnight Commander	8
3.1.2	Connecting an External File	14
3.2	Self-Study Task	18
4	Conclusions	25
	References	26

Список иллюстраций

3.1	Opening Midnight Commander	8
3.2	Midnight Commander Interface	9
3.3	Opened Directory arch-pc	9
3.4	Creating Working Subdirectory	10
3.5	Creating a File in Midnight Commander	11
3.6	Editing the File in Midnight Commander	12
3.7	Checking Saved Changes	13
3.8	Running the Modified Program	14
3.9	Copying the File to the Working Directory	15
3.10	Changing the Program	16
3.11	Launching the Modified Program	17
3.12	Running the Modified Program with Different Subroutine	18
3.13	Editing the Copy	19
3.14	Running the Program	20
3.15	Editing a copy	22
3.16	Running my program	23

Список таблиц

1 Objective of the Work

The objective of this laboratory work is to acquire practical skills in using Midnight Commander and to learn the basic instructions of the assembly language: mov and int.

2 Task

1. Basics of working with Midnight Commander
2. Structure of a program written in NASM assembly language
3. Including an external file
4. Completing the assignments for self-study

3 Theoretical Introduction

Midnight Commander (or simply mc) is a program that allows users to browse directory structures and perform basic file system management operations. Thus, mc functions as a file manager. Midnight Commander makes working with files more convenient and visually understandable.

A program written in the NASM assembly language typically consists of three sections: the program code section (`SECTION .text`), the initialized data section (known at compile time) (`SECTION .data`), and the uninitialized data section (which is allocated memory at compile time but assigned values during program execution) (`SECTION .bss`).

To declare initialized data in the `.data` section, the directives `DB`, `DW`, `DD`, `DQ`, and `DT` are used, which reserve memory and specify which values should be stored in that memory:

- `DB` (define byte) — defines a variable of 1 byte;
- `DW` (define word) — defines a variable of 2 bytes (word);
- `DD` (define double word) — defines a variable of 4 bytes (double word);
- `DQ` (define quad word) — defines a variable of 8 bytes (quad word);
- `DT` (define ten bytes) — defines a variable of 10 bytes.

3.1 Conducting the Laboratory Work

3.1.1 Basics of Working with Midnight Commander

By entering the appropriate command in the terminal (Figure -fig. 3.1), I open Midnight Commander (Figure -fig. 3.2).

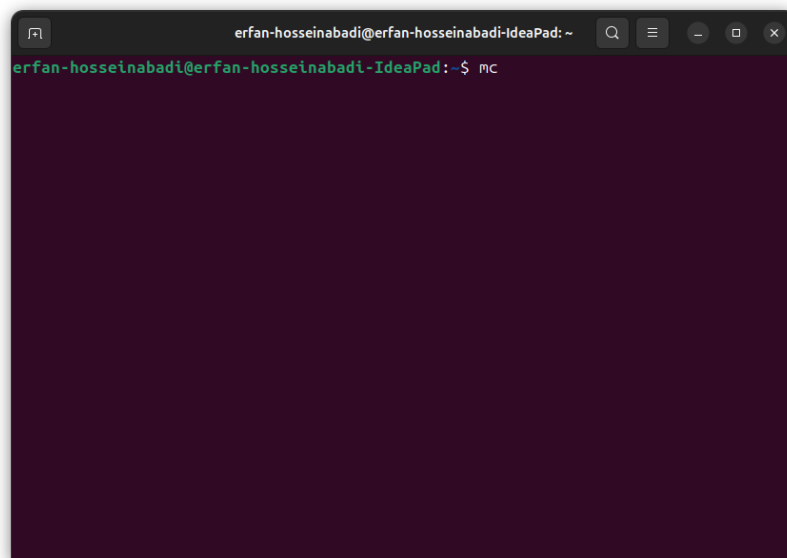


Рис. 3.1: Opening Midnight Commander

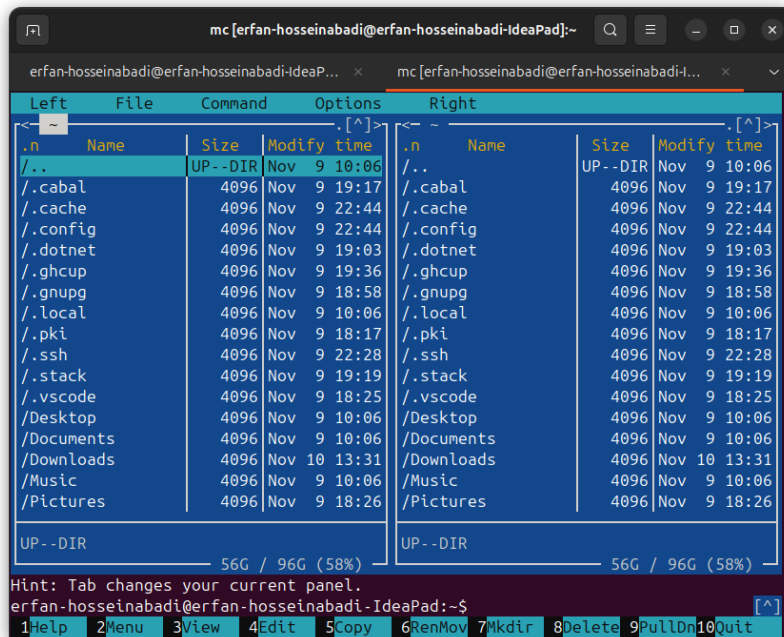


Рис. 3.2: Midnight Commander Interface

I navigate to the directory created in the previous laboratory work (Figure -fig. 3.3).

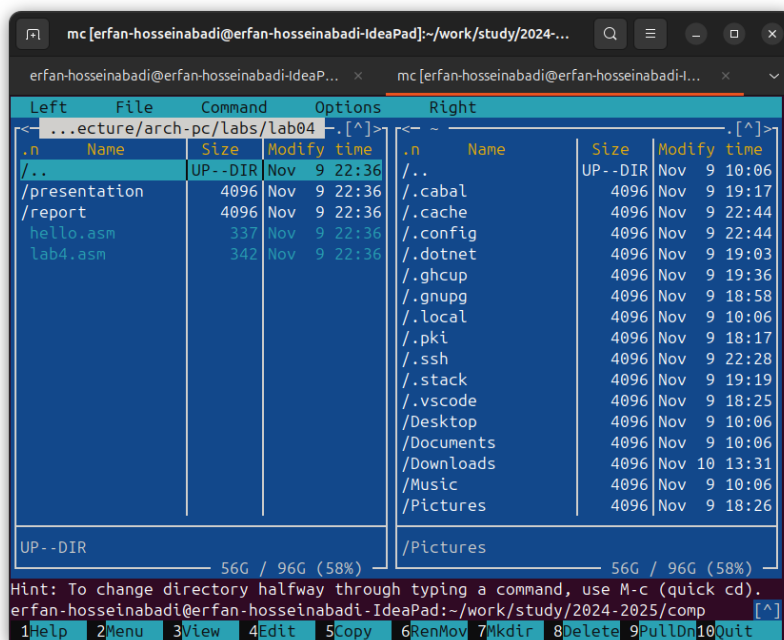


Рис. 3.3: Opened Directory arch-pc

Using the function key, I create a subdirectory lab05, where I will work (Figure - fig. 3.4).

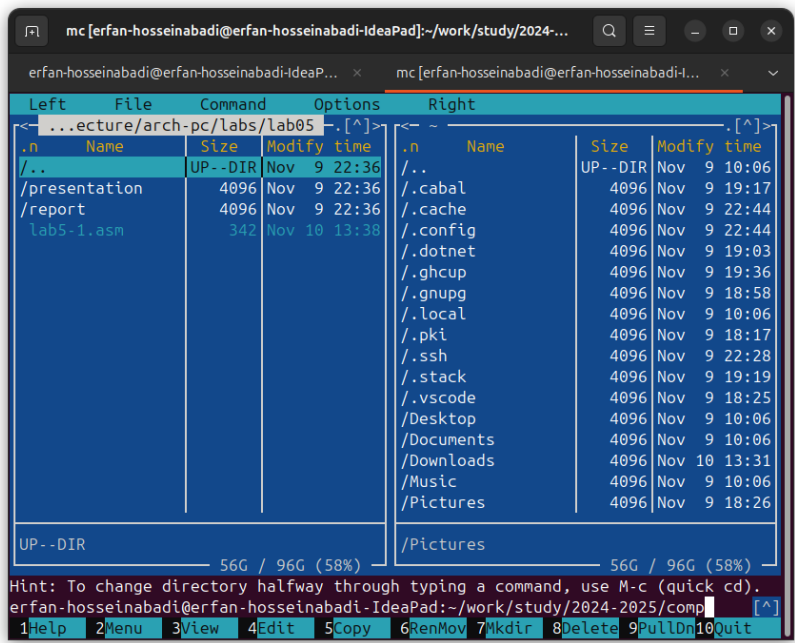


Рис. 3.4: Creating Working Subdirectory

In the input line, I enter the command touch and create a file (Figure -fig. 3.5).

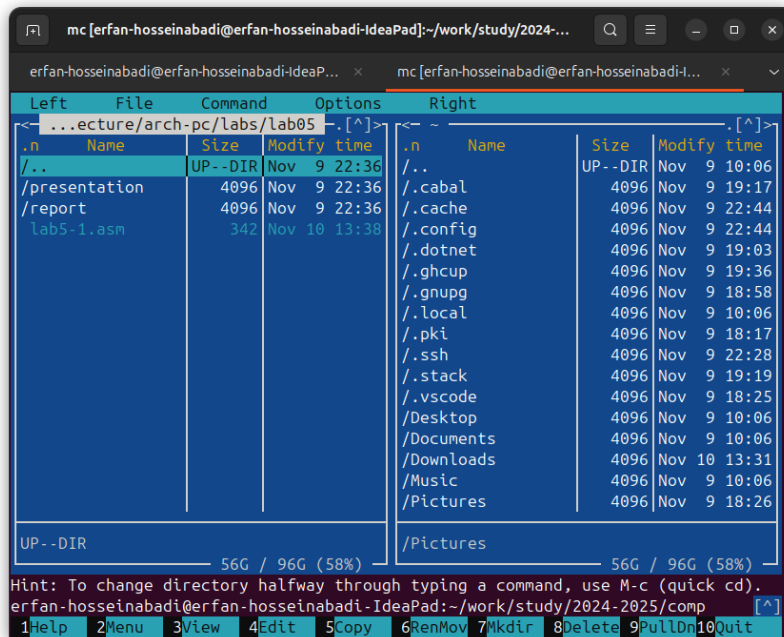
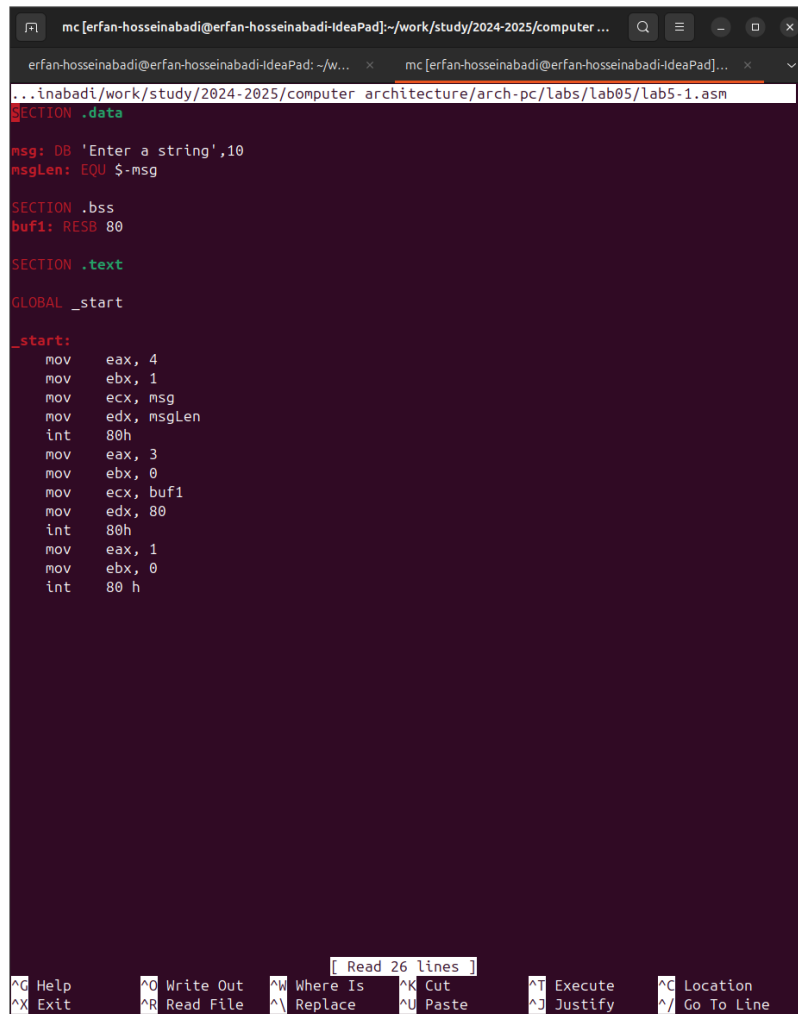


Рис. 3.5: Creating a File in Midnight Commander

I use F4 to open the newly created file and enter the code from the listing (Figure -fig. 3.6).



The screenshot shows the Midnight Commander (mc) interface. The top window title is 'mc [erfan-hosseiniabadi@erfan-hosseiniabadi-IdeaPad]: ~/work/study/2024-2025/computer ...'. The bottom window title is 'erfan-hosseiniabadi@erfan-hosseiniabadi-IdeaPad: ~/w...'. The current file being edited is '...inabadi/work/study/2024-2025/computer architecture/arch-pc/labs/lab05/lab5-1.asm'. The file content is as follows:

```
SECTION .data
msg: DB 'Enter a string',10
msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80

SECTION .text

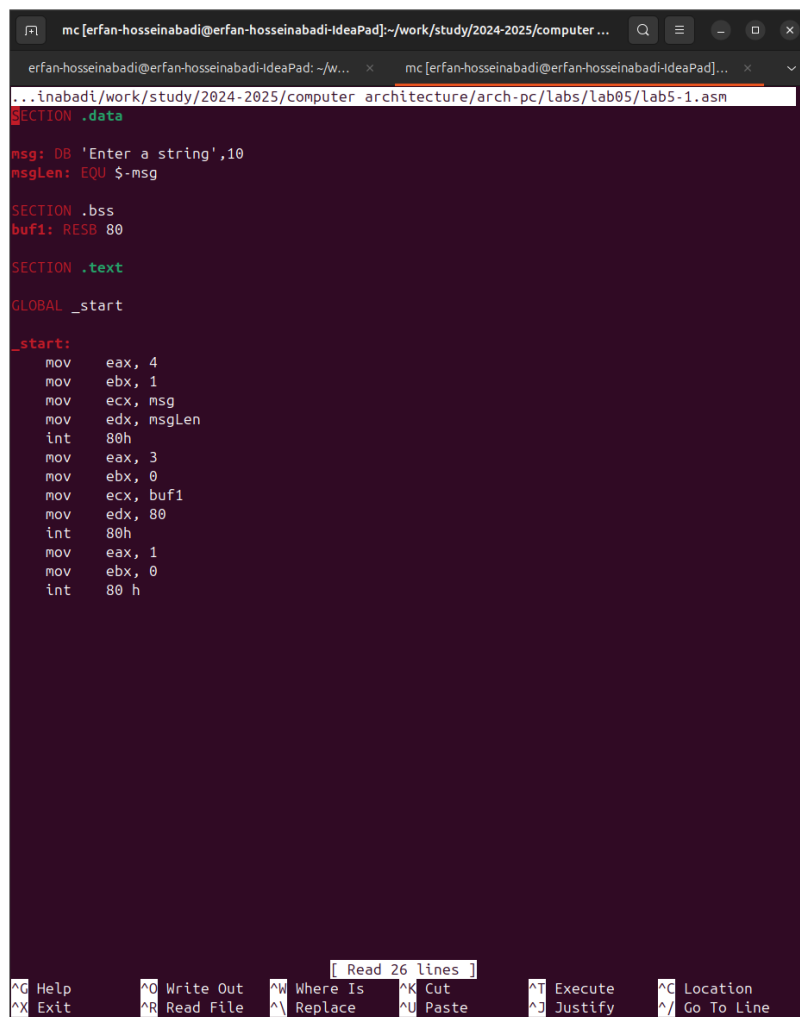
GLOBAL _start

_start:
    mov     eax, 4
    mov     ebx, 1
    mov     ecx, msg
    mov     edx, msgLen
    int     80h
    mov     eax, 3
    mov     ebx, 0
    mov     ecx, buf1
    mov     edx, 80
    int     80h
    mov     eax, 1
    mov     ebx, 0
    int     80 h
```

The status bar at the bottom shows the following keyboard shortcuts: ^G Help, ^X Exit, ^O Write Out, ^R Read File, ^W Where Is, ^M Replace, ^K Cut, ^U Paste, ^T Execute, ^J Justify, ^C Location, and ^_ Go To Line. A status message '[Read 26 lines]' is also visible.

Рис. 3.6: Editing the File in Midnight Commander

I check the saved changes using the F3 key (Figure -fig. 3.7).



```
mc [erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad]:~/work/study/2024-2025/computer ...
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad: ~/w... x mc [erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad]...
...inabadi/work/study/2024-2025/computer architecture/arch-pc/labs/lab05/lab5-1.asm
SECTION .data
msg: DB 'Enter a string',10
msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80

SECTION .text

GLOBAL _start

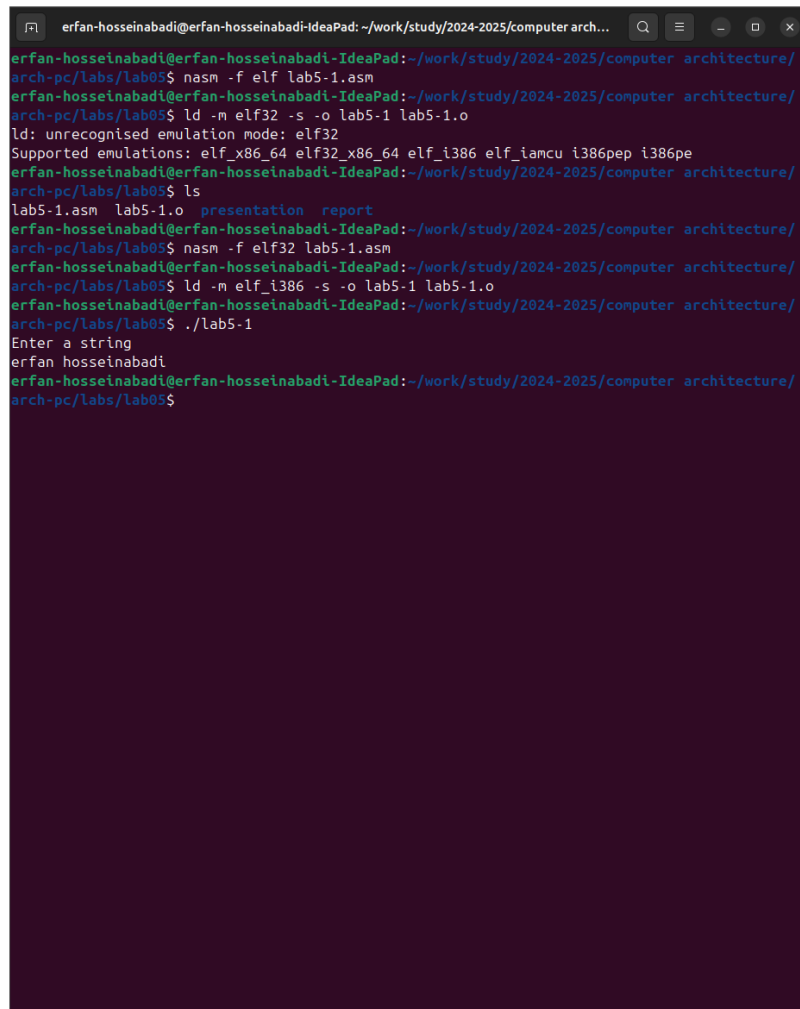
_start:
    mov     eax, 4
    mov     ebx, 1
    mov     ecx, msg
    mov     edx, msgLen
    int     80h
    mov     eax, 3
    mov     ebx, 0
    mov     ecx, buf1
    mov     edx, 80
    int     80h
    mov     eax, 1
    mov     ebx, 0
    int     80 h
```

[Read 26 lines]

^G Help ^O Write Out ^M Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^_ Go To Line

Рис. 3.7: Checking Saved Changes

I compile and execute the modified file (Figure -fig. 3.8).



```
erfan-hossebabadi@erfan-hossebabadi-IdeaPad: ~/work/study/2024-2025/computer arch...
erfan-hossebabadi@erfan-hossebabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab05$ nasm -f elf lab5-1.asm
erfan-hossebabadi@erfan-hossebabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab05$ ld -m elf32 -s -o lab5-1 lab5-1.o
ld: unrecognised emulation mode: elf32
Supported emulations: elf_x86_64 elf32_x86_64 elf_i386 elf_i386pe i386pep i386pe
erfan-hossebabadi@erfan-hossebabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab05$ ls
lab5-1.asm lab5-1.o presentation report
erfan-hossebabadi@erfan-hossebabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab05$ nasm -f elf32 lab5-1.asm
erfan-hossebabadi@erfan-hossebabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab05$ ld -m elf_i386 -s -o lab5-1 lab5-1.o
erfan-hossebabadi@erfan-hossebabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab05$ ./lab5-1
Enter a string
erfan hossebabadi
erfan-hossebabadi@erfan-hossebabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab05$
```

Рис. 3.8: Running the Modified Program

3.1.2 Connecting an External File

I save the file downloaded from TUIS to a shared folder on my computer, then in the virtual machine, I go to the directory of the shared folder, copy the file to the working subdirectory (Figure -fig. 3.9).

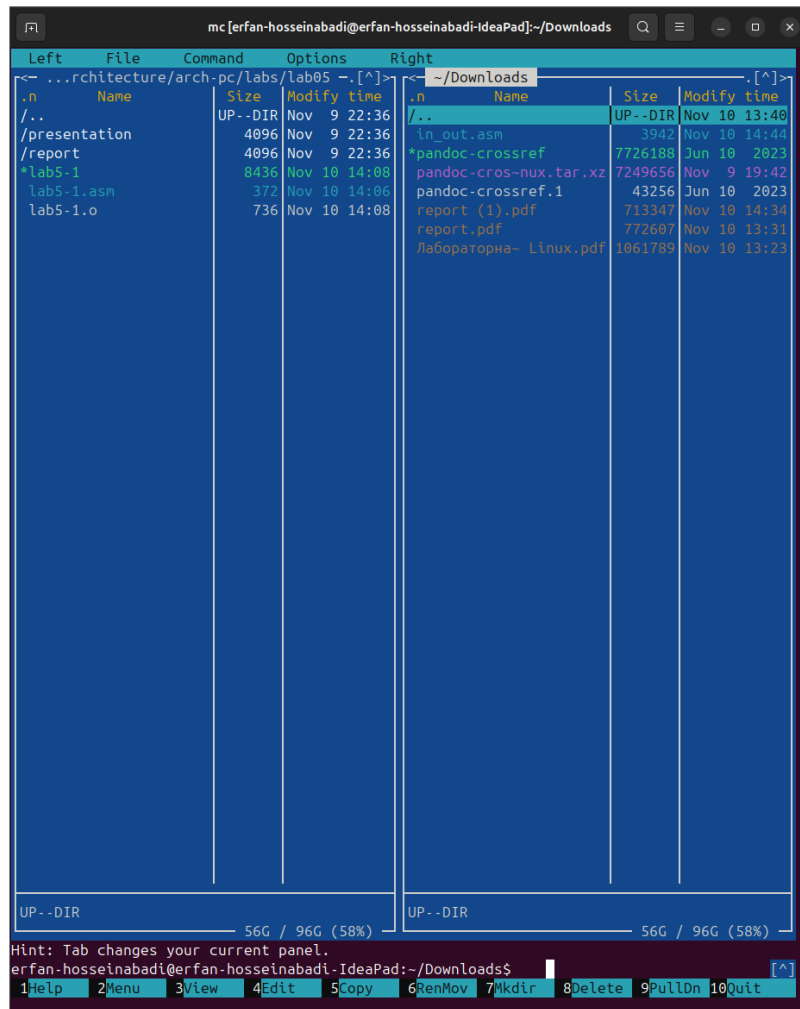
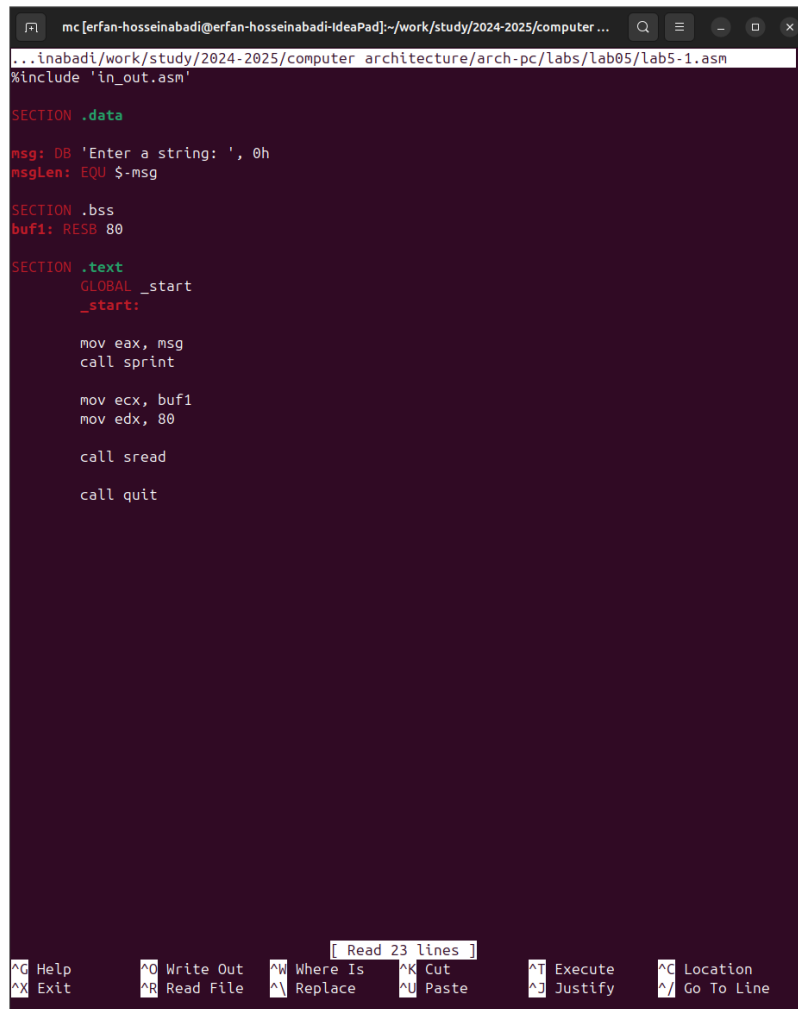


Рис. 3.9: Copying the File to the Working Directory

I include subroutines from the included file in the copy of the file (Figure -fig. 3.10).



```
mc [erfan-hosseiniabadi@erfan-hosseiniabadi-IdeaPad]:~/work/study/2024-2025/computer ...
...inabadi/work/study/2024-2025/computer architecture/arch-pc/labs/lab05/lab5-1.asm
#include 'in_out.asm'

SECTION .data
msg: DB 'Enter a string: ', 0h
msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80

SECTION .text
GLOBAL _start
_start:

    mov eax, msg
    call sprint

    mov ecx, buf1
    mov edx, 80

    call sread

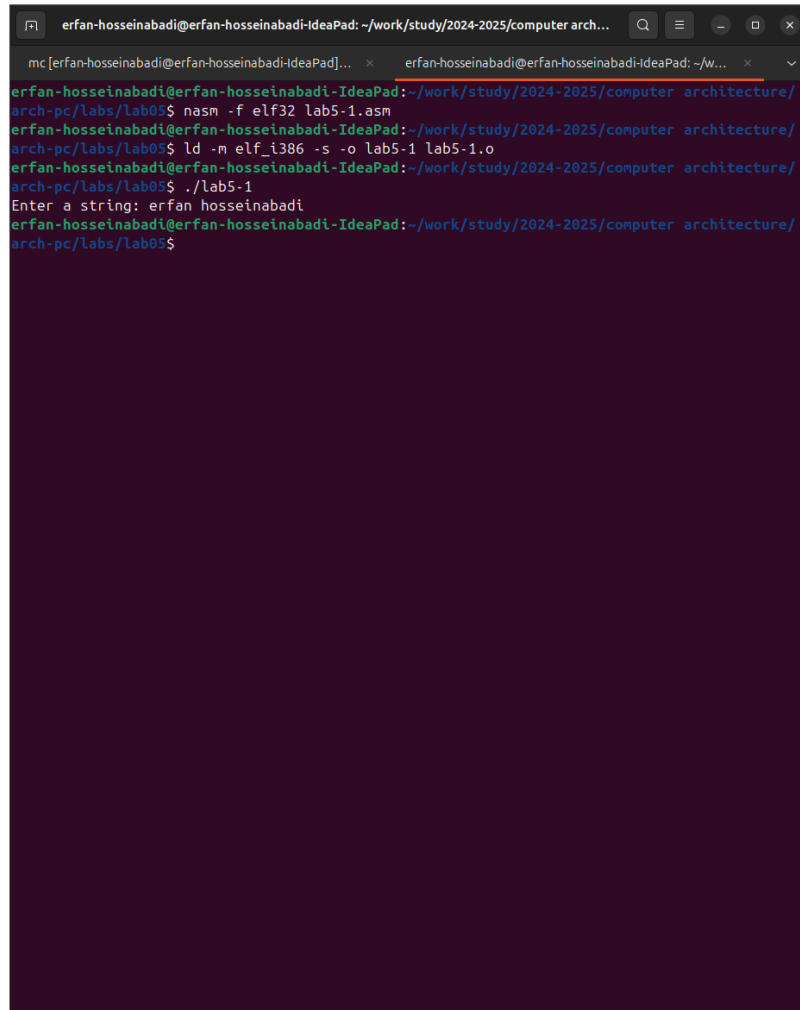
    call quit
```

[Read 23 lines]

^G Help	^O Write Out	^W Where Is	^K Cut	^T Execute	^C Location
^X Exit	^R Read File	^_ Replace	^U Paste	^J Justify	^_ Go To Line

Рис. 3.10: Changing the Program

I translate, compose, and launch the program with the included file (Figure -fig. 3.11).

A terminal window with a dark background and light-colored text. The window title is "erfan-hosseiniabadi@erfan-hosseiniabadi-IdeaPad: ~/work/study/2024-2025/computer arch...". The terminal shows the following commands and output:

```
erfan-hosseiniabadi@erfan-hosseiniabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab0$ nasm -f elf32 lab5-1.asm
erfan-hosseiniabadi@erfan-hosseiniabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab0$ ld -m elf_i386 -s -o lab5-1 lab5-1.o
erfan-hosseiniabadi@erfan-hosseiniabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab0$ ./lab5-1
Enter a string: erfan hosseinabadi
erfan-hosseiniabadi@erfan-hosseiniabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab0$
```

Рис. 3.11: Launching the Modified Program

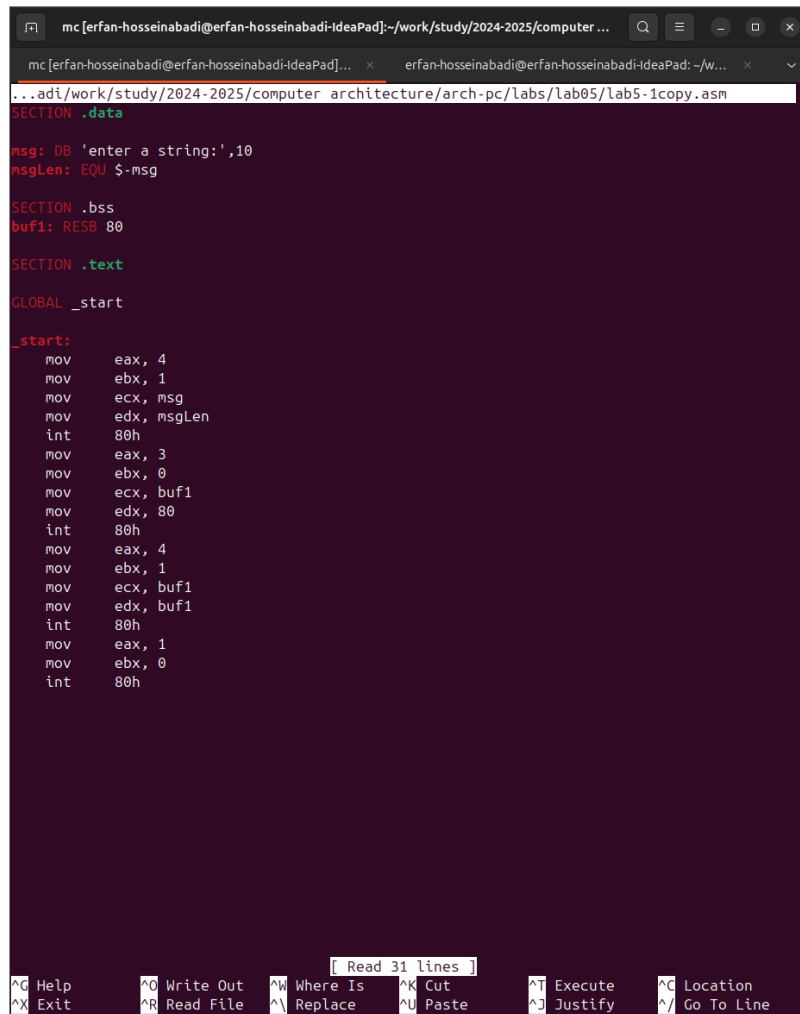
I edit the file and replace the `sprintf` subroutine with `sprint`. The difference between the two subroutines is that the second one prompts the input on the same line (Figure -fig. 3.12).

```
erfan-hossebabadi@erfan-hossebabadi-IdeaPad: ~/work/study/2024-2025/computer arch...
mc [erfan-hossebabadi@erfan-hossebabadi-IdeaPad]... x erfan-hossebabadi@erfan-hossebabadi-IdeaPad: ~/w...
erfan-hossebabadi@erfan-hossebabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab05$ nasm -f elf lab52.asm
nasm: fatal: unable to open input file 'lab52.asm' No such file or directory
erfan-hossebabadi@erfan-hossebabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab05$ ls
in_out.asm lab5-1 lab5-1.asm lab5-1.o lab5-2.asm presentation report
erfan-hossebabadi@erfan-hossebabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab05$ nasm -f elf lab5-2.asm
erfan-hossebabadi@erfan-hossebabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab05$ ld -m elf_i386 -o lab5-22 lab-2.o
ld: cannot find lab-2.o: No such file or directory
erfan-hossebabadi@erfan-hossebabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab05$ ld -m elf_i386 -o lab5-22 lab5-2.o
erfan-hossebabadi@erfan-hossebabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab05$ ./lab4-22
bash: ./lab4-22: No such file or directory
erfan-hossebabadi@erfan-hossebabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab05$ ./lab5-22
Enter a string
erfan hosseinabadi
erfan-hossebabadi@erfan-hossebabadi-IdeaPad:~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab05$
```

Рис. 3.12: Running the Modified Program with Different Subroutine

3.2 Self-Study Task

I create a copy of lab5-1.asm, editing it so that the string I entered from the keyboard is displayed at the end (Figure -fig. 3.13).



```
mc [erfan-hosseiniabadi@erfan-hosseiniabadi-IdeaPad]:~/work/study/2024-2025/computer ...
...adi/work/study/2024-2025/computer architecture/arch-pc/labs/lab05/lab5-1copy.asm
SECTION .data
msg: DB 'enter a string:',10
msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80

SECTION .text

GLOBAL _start

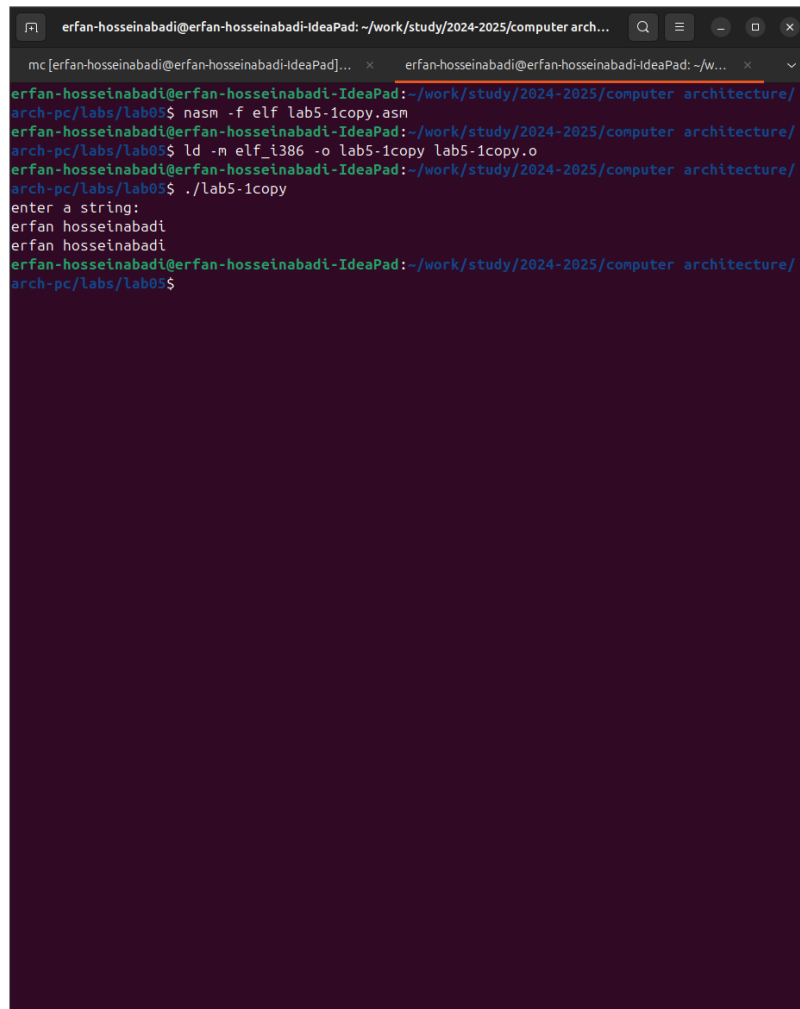
_start:
    mov     eax, 4
    mov     ebx, 1
    mov     ecx, msg
    mov     edx, msgLen
    int     80h
    mov     eax, 3
    mov     ebx, 0
    mov     ecx, buf1
    mov     edx, 80
    int     80h
    mov     eax, 4
    mov     ebx, 1
    mov     ecx, buf1
    mov     edx, buf1
    int     80h
    mov     eax, 1
    mov     ebx, 0
    int     80h
```

[Read 31 lines]

^G Help ^O Write Out ^W Where Is ^K Cut ^T Execute ^C Location
^X Exit ^R Read File ^\ Replace ^U Paste ^J Justify ^_ Go To Line

Рис. 3.13: Editing the Copy

I translate, compose, and run my program (Figure -fig. 3.14).



```
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad: ~/work/study/2024-2025/computer architecture/
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad:~/work/study/2024-2025/computer architecture/arch-pc/labs/lab05$ nasm -f elf lab5-1copy.asm
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad:~/work/study/2024-2025/computer architecture/arch-pc/labs/lab05$ ld -m elf_i386 -o lab5-1copy lab5-1copy.o
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad:~/work/study/2024-2025/computer architecture/arch-pc/labs/lab05$ ./lab5-1copy
enter a string:
erfan hosseinabadi
erfan hosseinabadi
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad:~/work/study/2024-2025/computer architecture/arch-pc/labs/lab05$
```

Рис. 3.14: Running the Program

Here is the code:

```
SECTION .data
msg: DB 'write a string:', 10
msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80

SECTION .text
```

```
GLOBAL _start
```

```
_start:
```

```
    mov     eax, 4
    mov     ebx, 1
    mov     ecx, msg
    mov     edx, msgLen
    int     80h
```

```

    mov     eax, 3
    mov     ebx, 0
    mov     ecx, buf1
    mov     edx, 80
    int     80h
```

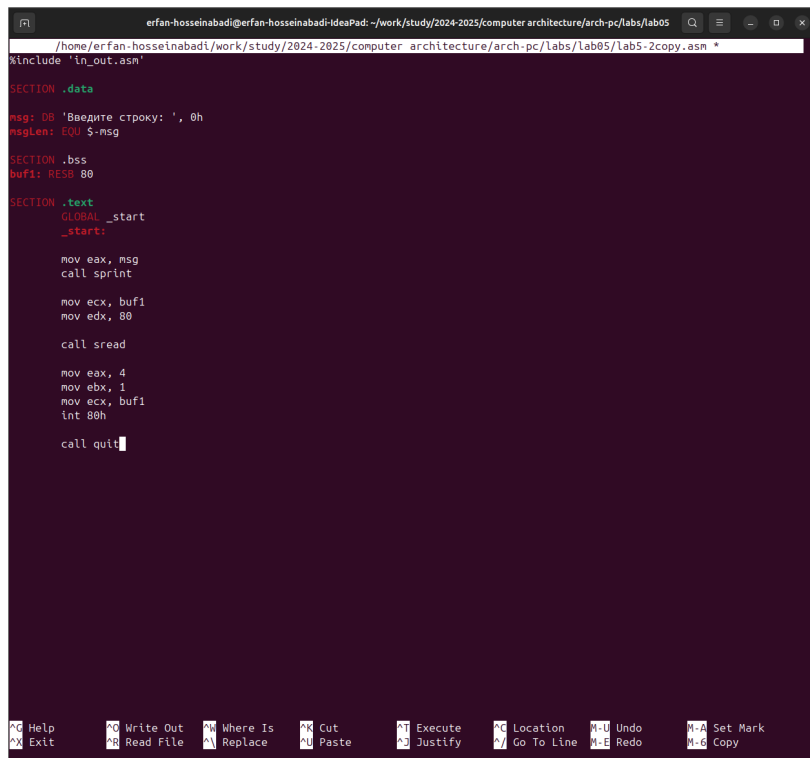
```

    mov     eax, 4
    mov     ebx, 1
    mov     ecx, buf1
    mov     edx, buf1
    int     80h
```

```

    mov     eax, 1
    mov     ebx, 0
    int     80h
```

I create a copy of lab5-2.asm, edit it so that the line I entered from the keyboard is displayed at the end (Fig. -fig. 3.15).



```
erfan-hosseiniabadi@erfan-hosseiniabadi-ideaPad: ~/work/study/2024-2025/computer architecture/arch-pc/labs/lab05
/home/erfan-hosseiniabadi/work/study/2024-2025/computer architecture/arch-pc/labs/lab05/lab5-2copy.asm *
#include "in_out.asm"

SECTION .data
msg: DB "Введите строку: ", 0h
msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80

SECTION .text
GLOBAL _start
_start:

    mov eax, msg
    call sprint

    mov ecx, buf1
    mov edx, 80

    call sread

    mov eax, 4
    mov ebx, 1
    mov ecx, buf1
    int 80h

    call quit
```

Рис. 3.15: Editing a copy

I translate, compose and run my program (Fig. -fig. 3.16).

```
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad: ~/work/study/2024-2025/computer arch...
mc [erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad]... x erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad: ~/w... x v
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad: ~/work/study/2024-2025/computer architectureeee
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad: ~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab0$ nasm -f elf lab5-2copy.asm
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad: ~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab0$ ld -m elf
ld: unrecognized emulation mode: elf
Supported emulations: elf_x86_64 elf32_x86_64 elf_i386 elf_iamcu i386pep i386pe
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad: ~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab0$
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad: ~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab0$
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad: ~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab0$ ld -m elf_i386 -o lab5-2copy lab5-2copy.o
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad: ~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab0$ ./lab5-2copy
bash: ./lab5-2copy: No such file or directory
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad: ~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab0$ ls
in_out.asm  lab5-1copy  lab5-1.o  lab5-2copy  lab5-2.o
lab5-1      lab5-1copy.asm  lab5-22  lab5-2copy.asm  presentation
lab5-1.asm  lab5-1copy.o  lab5-2.asm  lab5-2copy.o  report
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad: ~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab0$ ./lab5-2copy
Введите строку: erfan hosseinabadi
erfan hosseinabadi
erfan-hosseinabadi@erfan-hosseinabadi-IdeaPad: ~/work/study/2024-2025/computer architecture/
arch-pc/labs/lab0$
```

Рис. 3.16: Running my program

code:

```
%include 'in_out.asm'
```

```
SECTION .data
```

```
msg: DB 'write a string: ', 0h
```

```
msgLen: EQU $-msg
```

```
SECTION .bss
```

```
buf1: RESB 80
```

```
SECTION .text
```

```
GLOBAL _start
```

```
_start:
```

```
mov eax, msg
```

```
call sprint
```

```
mov ecx, buf1
```

```
mov edx, 80
```

```
call sread
```

```
mov eax, 4
```

```
mov ebx, 1
```

```
mov ecx, buf1
```

```
int 80h
```

```
call quit
```


4 Conclusions

During this lab I gained practical skills in working in Midnight Commander and also mastered the assembly language instructions `mov` and `int`.

References

1. sample
2. course on tuis
3. labrotary work №5
4. prograaming in nasmlanguage