

Отчёта по лабораторной работе 6

Основы работы с Midnight Commander (mc). Структура программы на языке ассемблера NASM

Кара-сал Эльдар Эдуардович

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	14

Список иллюстраций

2.1	Создание файлов в Midnight Commander	6
2.2	Редактирование файла 1 в Midnight Commander	7
2.3	Проверка программы 1	8
2.4	Файл in_out.asm	8
2.5	Редактирование файла 2 в Midnight Commander	9
2.6	Проверка программы 2	9
2.7	Редактирование файла 3 в Midnight Commander	10
2.8	Проверка программы 3	10
2.9	Редактирование файла 4 в Midnight Commander	11
2.10	Проверка программы 4	12
2.11	Редактирование файла 5 в Midnight Commander	13
2.12	Проверка программы 5	13

Список таблиц

1 Цель работы

Целью работы является приобретение практических навыков работы в Midnight Commander. Освоение инструкций языка ассемблера `mov` и `int`.

2 Выполнение лабораторной работы

1. Создадим новый подкаталог с именем lab06 и в нем файл lab6-1.asm. (рис. [2.1])

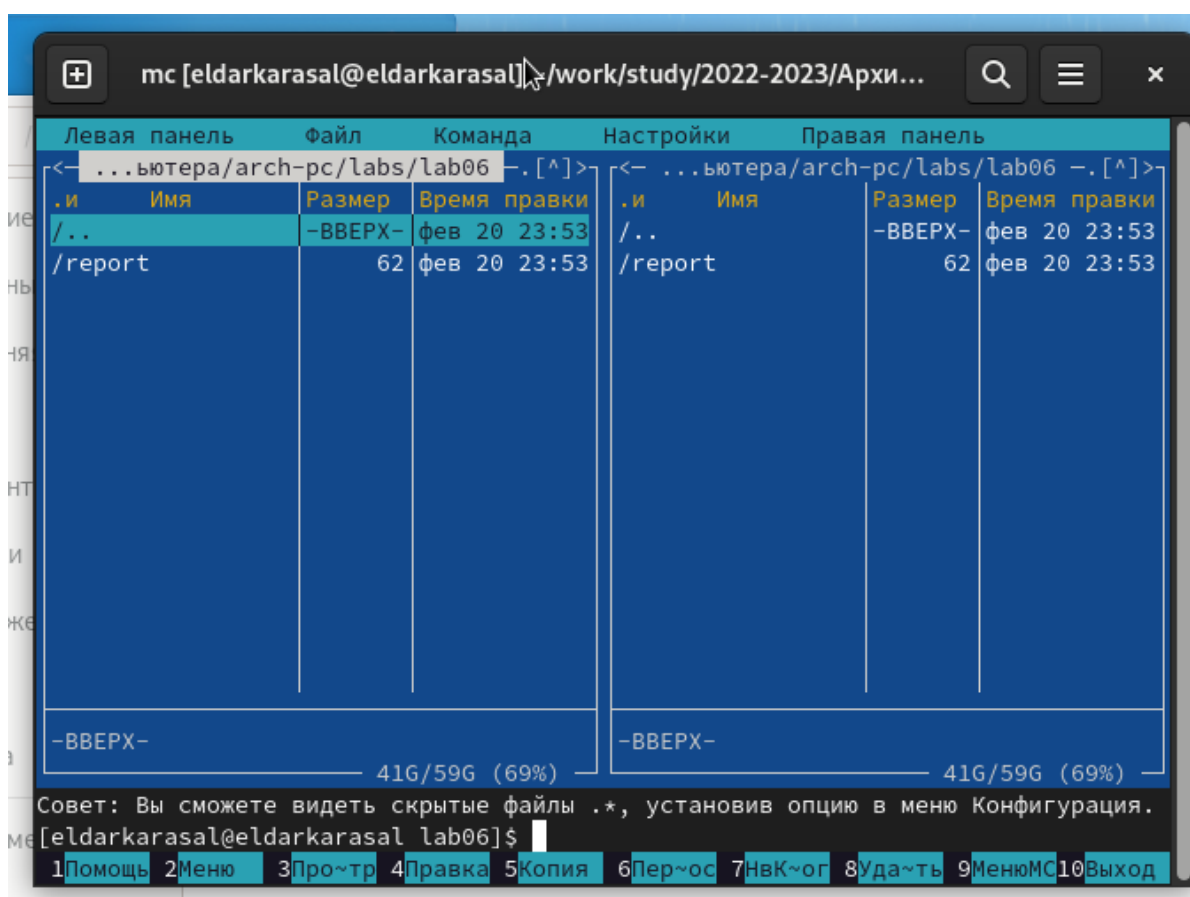
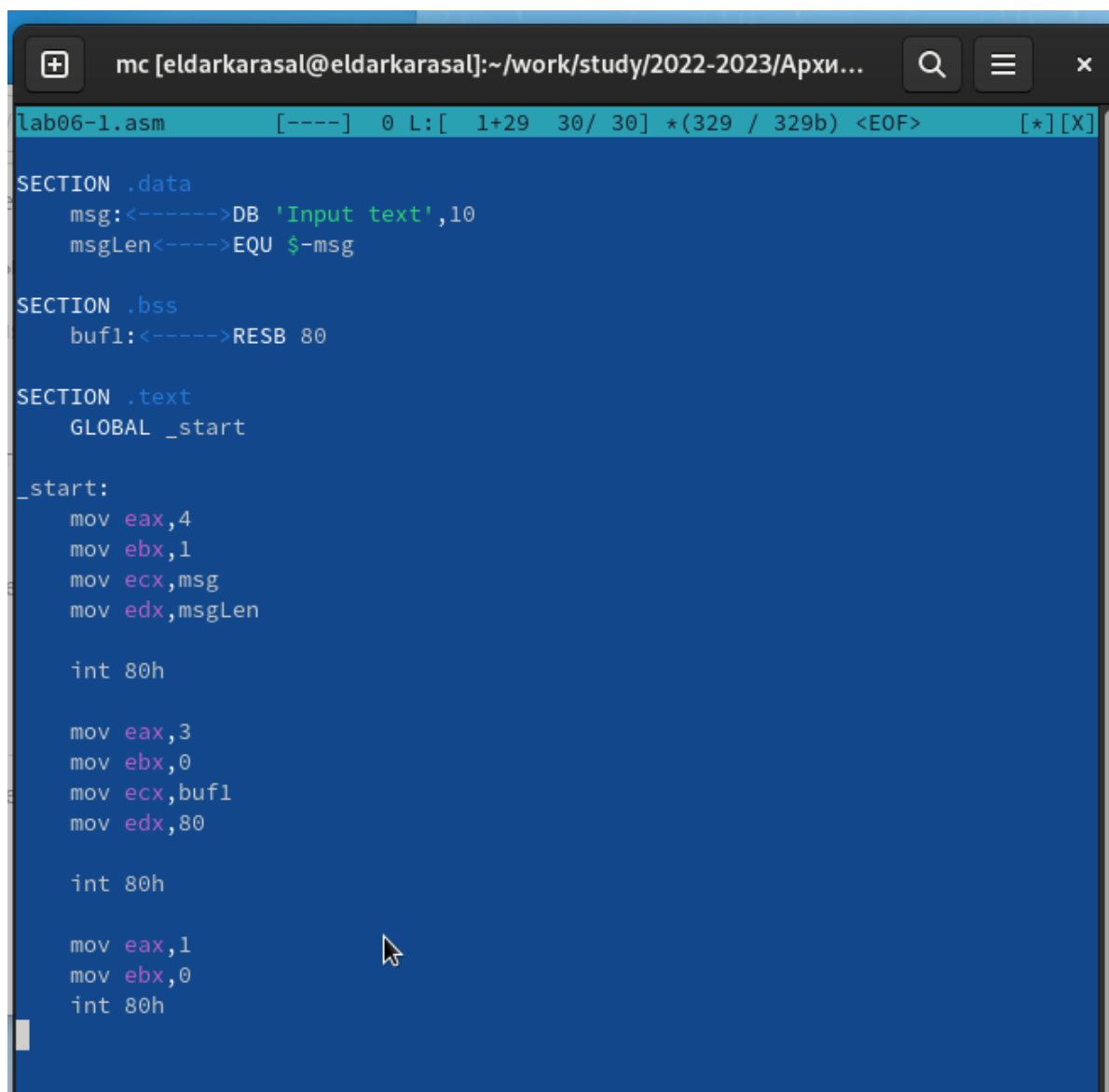


Рис. 2.1: Создание файлов в Midnight Commander

2. Введем в файл lab6-1.asm текст программы вывода сообщения на экран и

ввода строки с клавиатуры (Листинг 1.). Создадим исполняемый файл и проверим его работу. (рис. [2.2], [2.3])



The image shows a screenshot of the Midnight Commander (mc) file manager and editor. The title bar indicates the user is 'eldarkarasal' at a terminal, editing a file named 'lab06-1.asm' in the directory '~/.work/study/2022-2023/Архи...'. The status bar at the top shows the current line is 1 of 29, column 30, and the file size is 329 bytes. The main editing area contains assembly code for three sections: .data, .bss, and .text. The .data section defines a message 'Input text' of length 10. The .bss section reserves 80 bytes for a buffer. The .text section starts at address 0 and contains instructions for reading input from the user and writing it to the buffer. The code uses standard x86 assembly instructions like 'mov', 'int', and 'global'.

```
lab06-1.asm [----] 0 L: [ 1+29 30/ 30] *(329 / 329b) <EOF> [*] [X]

SECTION .data
    msg:<----->DB 'Input text',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 80h
```

Рис. 2.2: Редактирование файла 1 в Midnight Commander

```

eldarkarasal@eldarkarasal:~/work/study/2022-2023/Архитекту...
[eldarkarasal@eldarkarasal lab06]$ nasm -f elf lab06-1.asm
[eldarkarasal@eldarkarasal lab06]$ ld -m elf_i386 -o lab06-1 lab06-1.o
[eldarkarasal@eldarkarasal lab06]$ ./lab06-1
Input text
Eldar
[eldarkarasal@eldarkarasal lab06]$

```

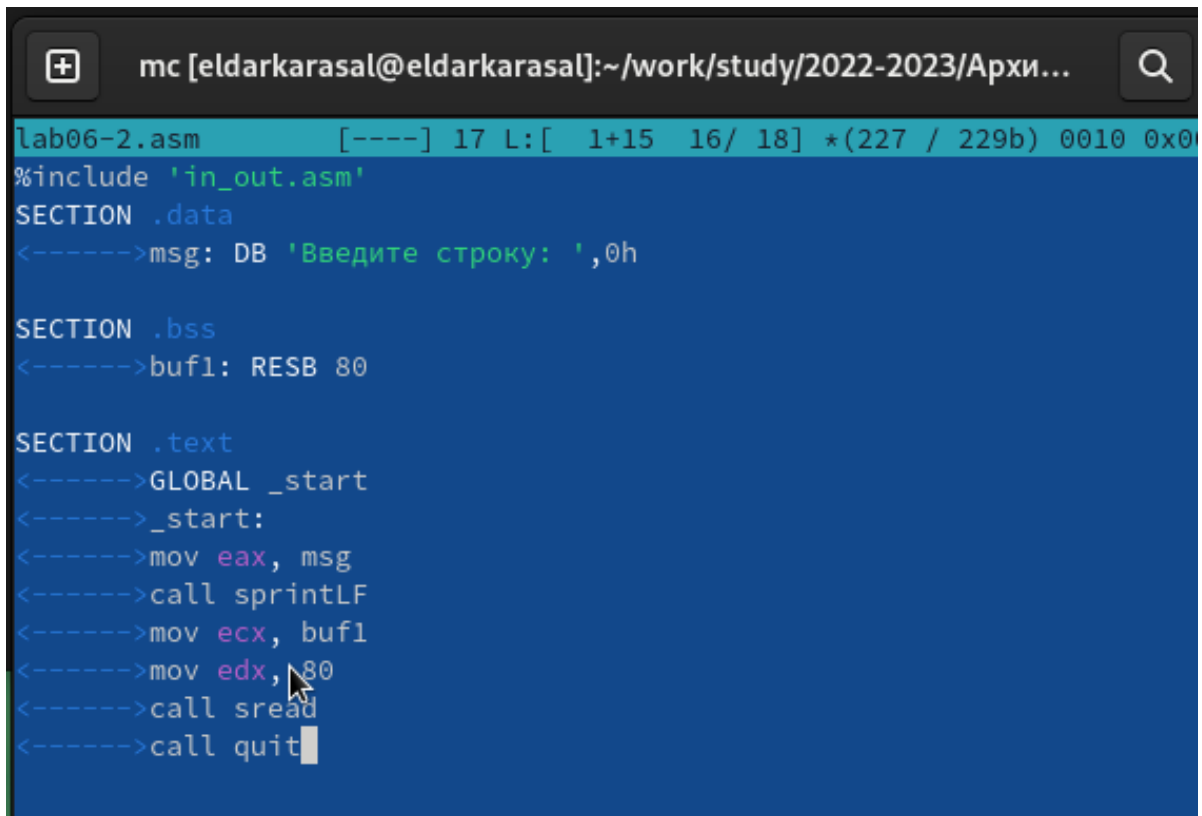
Рис. 2.3: Проверка программы 1

3. Скачали с туис доп файл, скопировали программу. (рис. [2.4])

Левая панель				Правая панель			
Имя	Размер	Время	Правки	Имя	Размер	Время	Правки
./..	-ВВЕРХ-	фев 20	23:53	./..	-ВВЕРХ-	фев 20	23:53
/report	62	фев 20	23:53	/report	62	фев 20	23:53
in_out.asm	3773	фев 21	11:27	in_out.asm	3773	фев 21	11:27
*lab06-1	8728	фев 21	11:27	*lab06-1	8728	фев 21	11:27
lab06-1.asm	329	фев 21	11:26	lab06-1.asm	329	фев 21	11:26
lab06-1.o	736	фев 21	11:27	lab06-1.o	736	фев 21	11:27
lab06-2.asm	229	фев 21	11:27	lab06-2.asm	229	фев 21	11:27

Рис. 2.4: Файл in_out.asm

4. Изменили код программы. (рис. [2.5], [2.6])



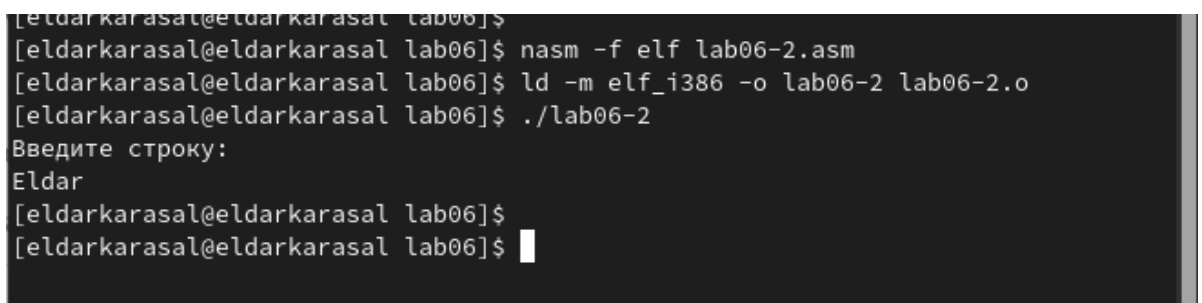
The screenshot shows the Midnight Commander (mc) interface. The title bar indicates the user is 'eldarkarasal' and the current directory is '~/work/study/2022-2023/Архи...'. The editor window displays the assembly file 'lab06-2.asm' at line 17. The code includes a data section with a message string, a bss section for a buffer, and a text section with assembly instructions for printing and reading input.

```
lab06-2.asm [----] 17 L: [ 1+15 16/ 18] *(227 / 229b) 0010 0x00
%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
<----->call quit
```

Рис. 2.5: Редактирование файла 2 в Midnight Commander

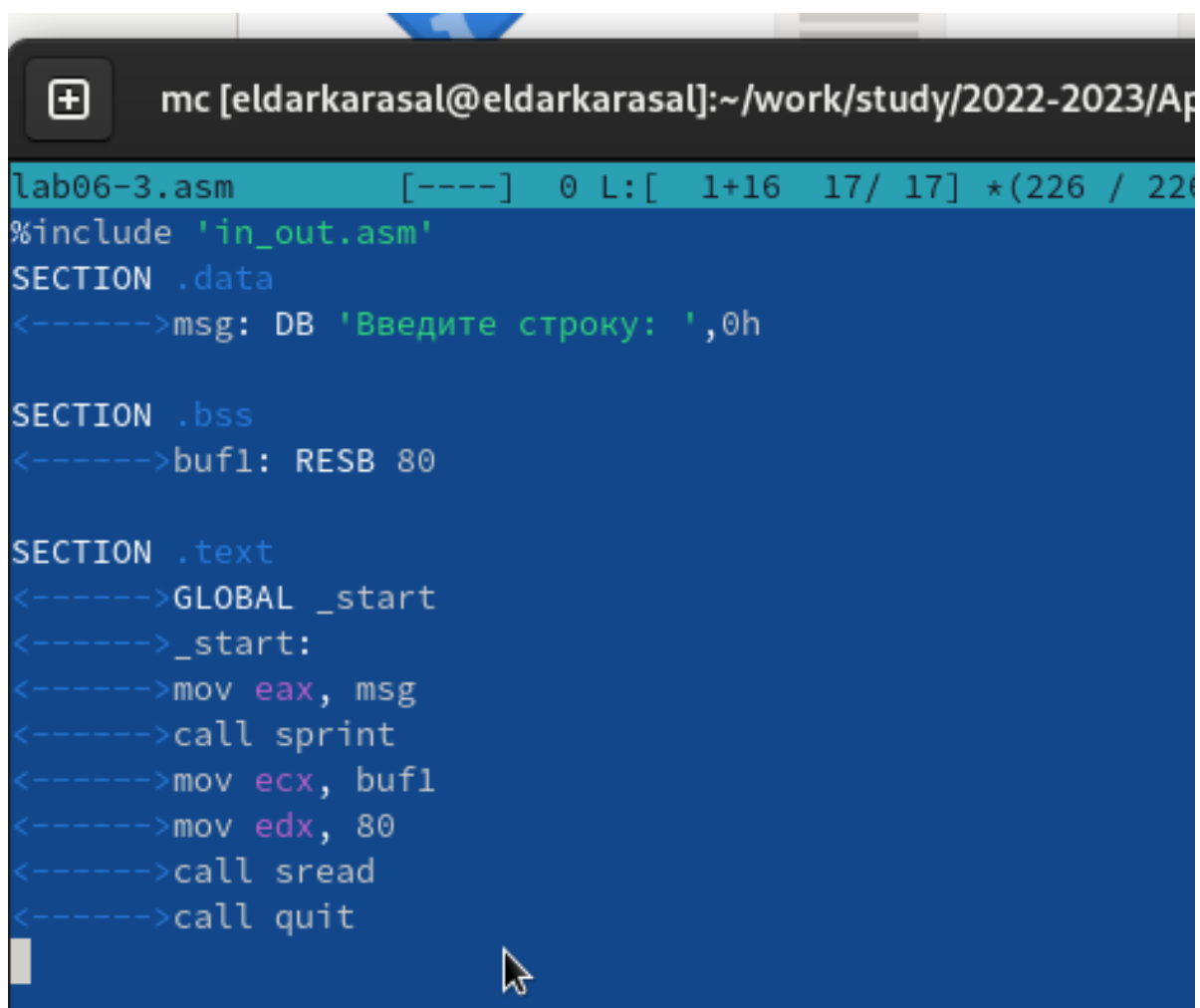


The screenshot shows a terminal window where the assembly program is compiled using nasm and ld, and then executed. The program prompts for input, and the user enters 'Eldar'.

```
[eldarkarasal@eldarkarasal lab06]$ nasm -f elf lab06-2.asm
[eldarkarasal@eldarkarasal lab06]$ ld -m elf_i386 -o lab06-2 lab06-2.o
[eldarkarasal@eldarkarasal lab06]$ ./lab06-2
Введите строку:
Eldar
[eldarkarasal@eldarkarasal lab06]$
[eldarkarasal@eldarkarasal lab06]$
```

Рис. 2.6: Проверка программы 2

5. Изменили вызов подпрограммы. Теперь ввод и вывод в одну строку. (рис. [2.7], [2.8])



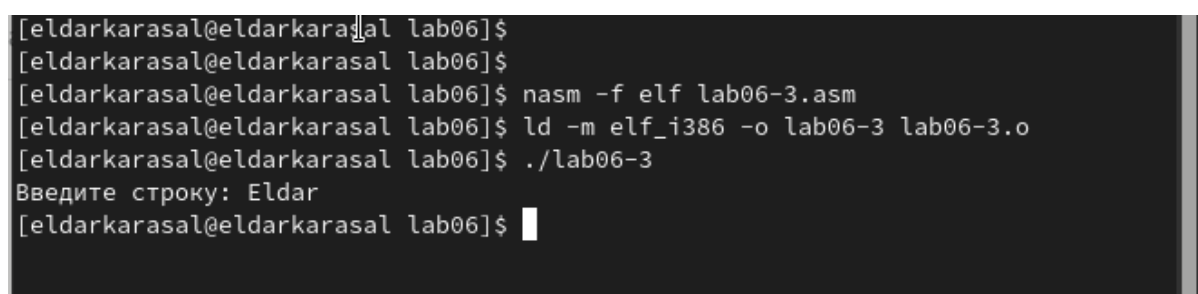
The screenshot shows the Midnight Commander file manager and editor. The title bar indicates the user is 'mc [eldarkarasal@eldarkarasal]:~/work/study/2022-2023/Ar'. The editor window displays the assembly file 'lab06-3.asm' with a status bar showing '[----] 0 L: [1+16 17/ 17] *(226 / 226)'. The code includes a directive to include 'in_out.asm', followed by three sections: .data, .bss, and .text. The .data section contains a message string 'Введите строку: ',0h. The .bss section reserves 80 bytes for a buffer named buf1. The .text section defines the _start function, which uses system calls to print the message, read input into the buffer, and then quit.

```
lab06-3.asm [----] 0 L: [ 1+16 17/ 17] *(226 / 226)
%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 sprint
<----->mov ecx, buf1
<----->mov edx, 80
<----->call sread
<----->call quit
```

Рис. 2.7: Редактирование файла 3 в Midnight Commander



The screenshot shows a terminal window with the following commands and output: the user navigates to the 'lab06' directory, uses 'nasm' to compile 'lab06-3.asm' into 'lab06-3.o', and then 'ld' to link it into an executable. Finally, the user runs the program, which prompts for input and the user enters 'Eldar'.

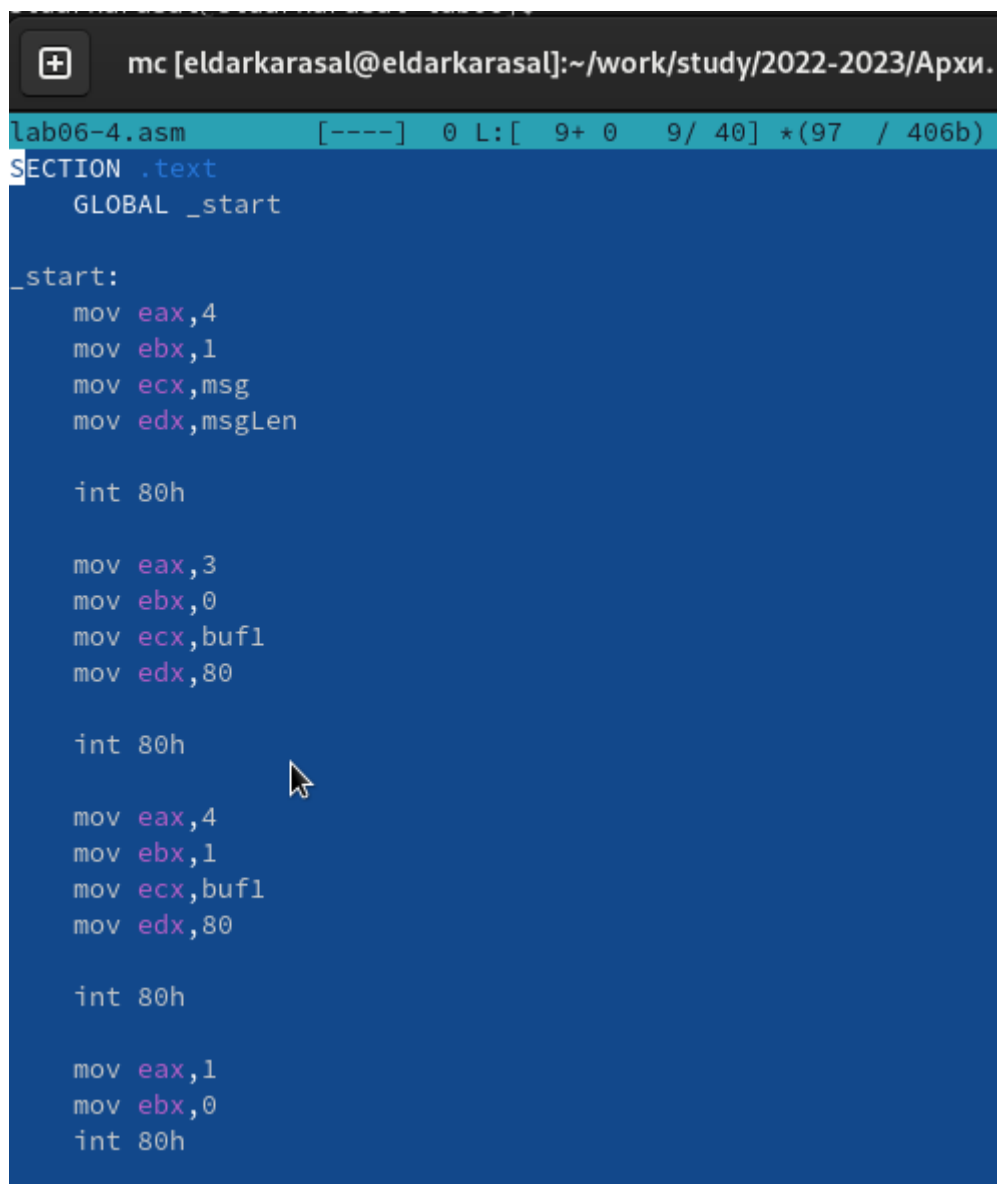
```
[eldarkarasal@eldarkarasal lab06]$
[eldarkarasal@eldarkarasal lab06]$
[eldarkarasal@eldarkarasal lab06]$ nasm -f elf lab06-3.asm
[eldarkarasal@eldarkarasal lab06]$ ld -m elf_i386 -o lab06-3 lab06-3.o
[eldarkarasal@eldarkarasal lab06]$ ./lab06-3
Введите строку: Eldar
[eldarkarasal@eldarkarasal lab06]$
```

Рис. 2.8: Проверка программы 3

6. Внесем изменения в программу (без использования внешнего файла

in_out.asm), так чтобы она работала по следующему алгоритму: (рис. [2.9], [2.10])

- вывести приглашение типа “Введите строку:”;
- ввести строку с клавиатуры;
- вывести введенную строку на экран.

A screenshot of the Midnight Commander file manager and editor. The title bar shows the user 'eldarkarasal' and the file path '~/work/study/2022-2023/Архи.'. The editor window displays the assembly file 'lab06-4.asm'. The code is as follows:

```
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,80

    int 80h

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

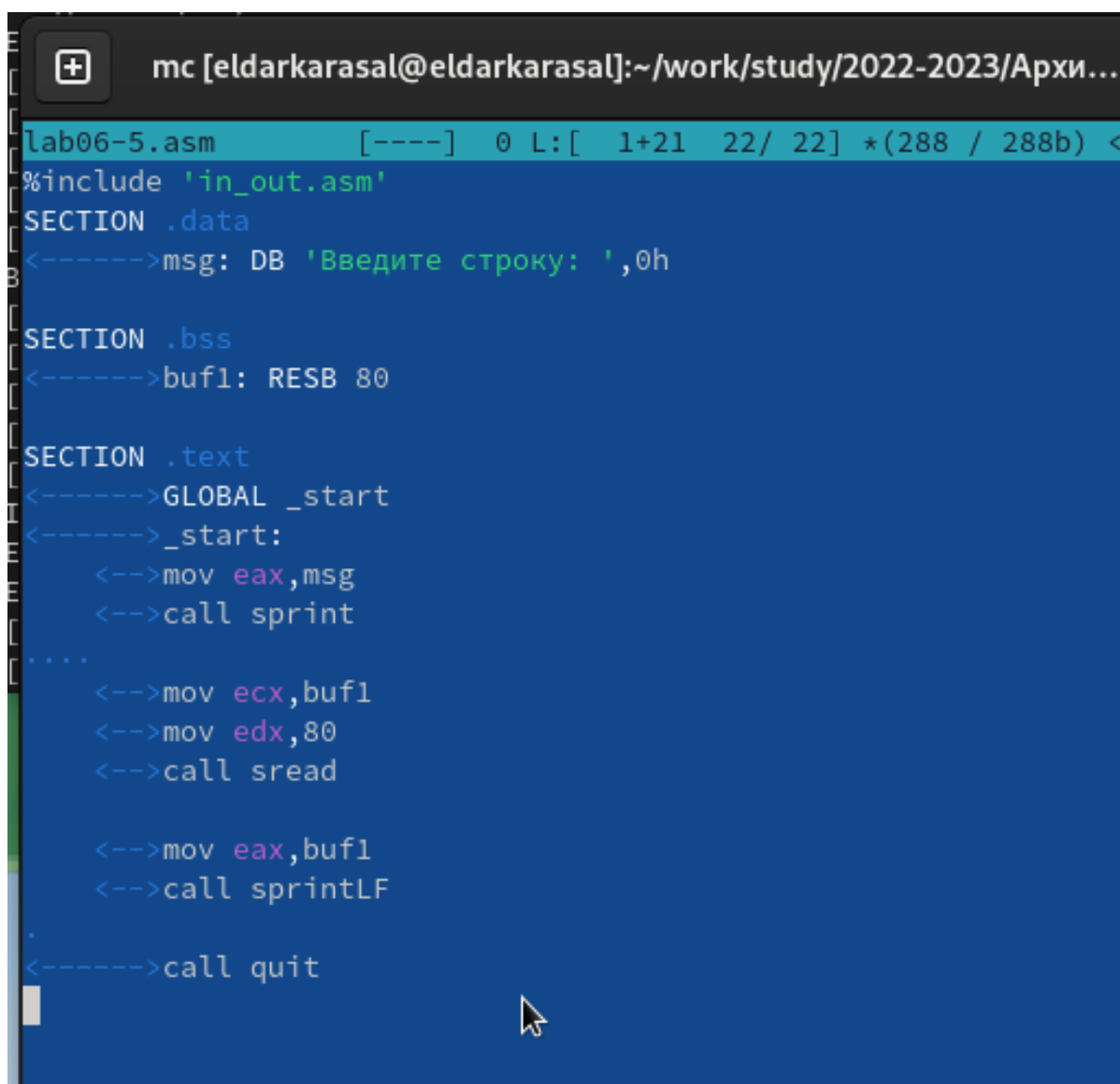
A mouse cursor is visible over the code.

Рис. 2.9: Редактирование файла 4 в Midnight Commander

```
[eldarkarasal@eldarkarasal lab06]$  
[eldarkarasal@eldarkarasal lab06]$ nasm -f elf lab06-4.asm  
[eldarkarasal@eldarkarasal lab06]$ ld -m elf_i386 -o lab06-4 lab06-4.o  
[eldarkarasal@eldarkarasal lab06]$ ./lab06-4  
Input text  
Eldar  
Eldar  
[eldarkarasal@eldarkarasal lab06]$  
[eldarkarasal@eldarkarasal lab06]$
```

Рис. 2.10: Проверка программы 4

7. Сделаем тоже самое с файлом in_out.asm (рис. [2.11], [2.12])



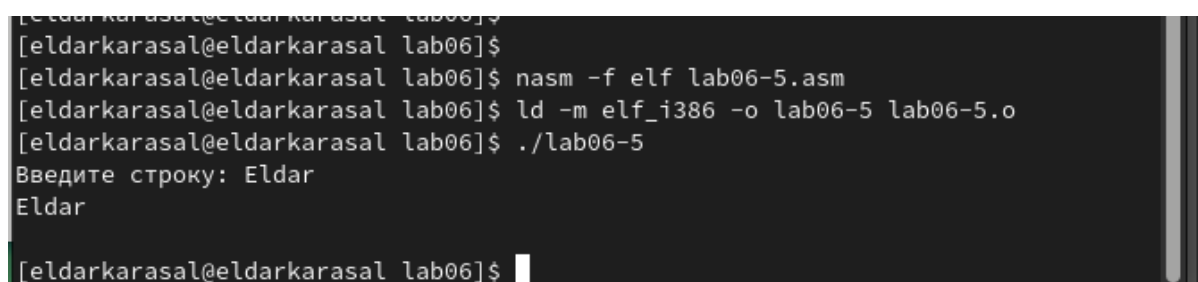
```
mc [eldarkarasal@eldarkarasal]:~/work/study/2022-2023/Архи...
lab06-5.asm [----] 0 L:[ 1+21 22/ 22] *(288 / 288b) <
%include 'in_out.asm'
SECTION .data
<----->msg: DB 'Введите строку: ',0h

SECTION .bss
<----->buf1: RESB 80

SECTION .text
<----->GLOBAL _start
<----->_start:
    <-->mov eax,msg
    <-->call sprint
    ....
    <-->mov ecx,buf1
    <-->mov edx,80
    <-->call sread

    <-->mov eax,buf1
    <-->call sprintLF
    .
<----->call quit
```

Рис. 2.11: Редактирование файла 5 в Midnight Commander



```
[eldarkarasal@eldarkarasal lab06]$
[eldarkarasal@eldarkarasal lab06]$ nasm -f elf lab06-5.asm
[eldarkarasal@eldarkarasal lab06]$ ld -m elf_i386 -o lab06-5 lab06-5.o
[eldarkarasal@eldarkarasal lab06]$ ./lab06-5
Введите строку: Eldar
Eldar
[eldarkarasal@eldarkarasal lab06]$
```

Рис. 2.12: Проверка программы 5

3 Выводы

Научились писать базовые ассемблерные программы. Освоили ассемблерные инструкции `mov` и `int`.