

# **Отчёт по лабораторной работе 5**

**Архитектура компьютера**

Мирзоян Давид Игнатович НБИбд-01-23

# Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	19

## Список иллюстраций

2.1	Создание каталога . . . . .	6
2.2	Создание файла lab05-1.asm . . . . .	7
2.3	Программа в файле lab05-1.asm . . . . .	8
2.4	Просмотр файла lab05-1.asm . . . . .	10
2.5	Сборка и проверка программы lab05-1.asm . . . . .	10
2.6	Копирование файла . . . . .	11
2.7	Программа в файле lab05-2.asm . . . . .	12
2.8	Сборка и проверка программы lab05-2.asm . . . . .	13
2.9	Программа в файле lab05-2.asm . . . . .	13
2.10	Сборка и проверка программы lab05-2.asm . . . . .	14
2.11	Программа в файле lab05-3.asm . . . . .	15
2.12	Сборка и проверка программы lab05-3.asm . . . . .	16
2.13	Программа в файле lab05-4.asm . . . . .	17
2.14	Сборка и проверка программы lab05-4.asm . . . . .	18

## **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

Открыл Midnight Commander

Перешел в каталог ~/work/arch-pc

Создал каталог lab05

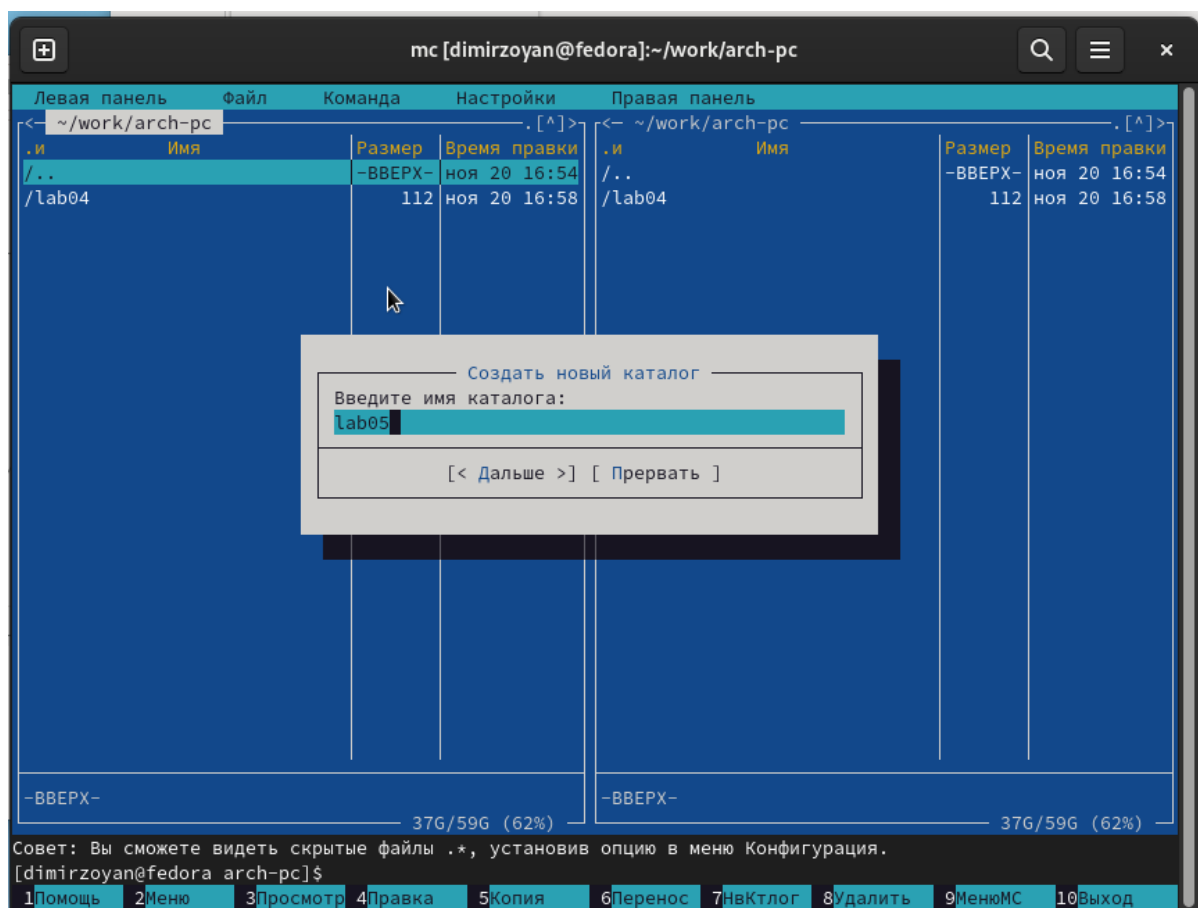


Рис. 2.1: Создание каталога

Создал файл lab05-1.asm

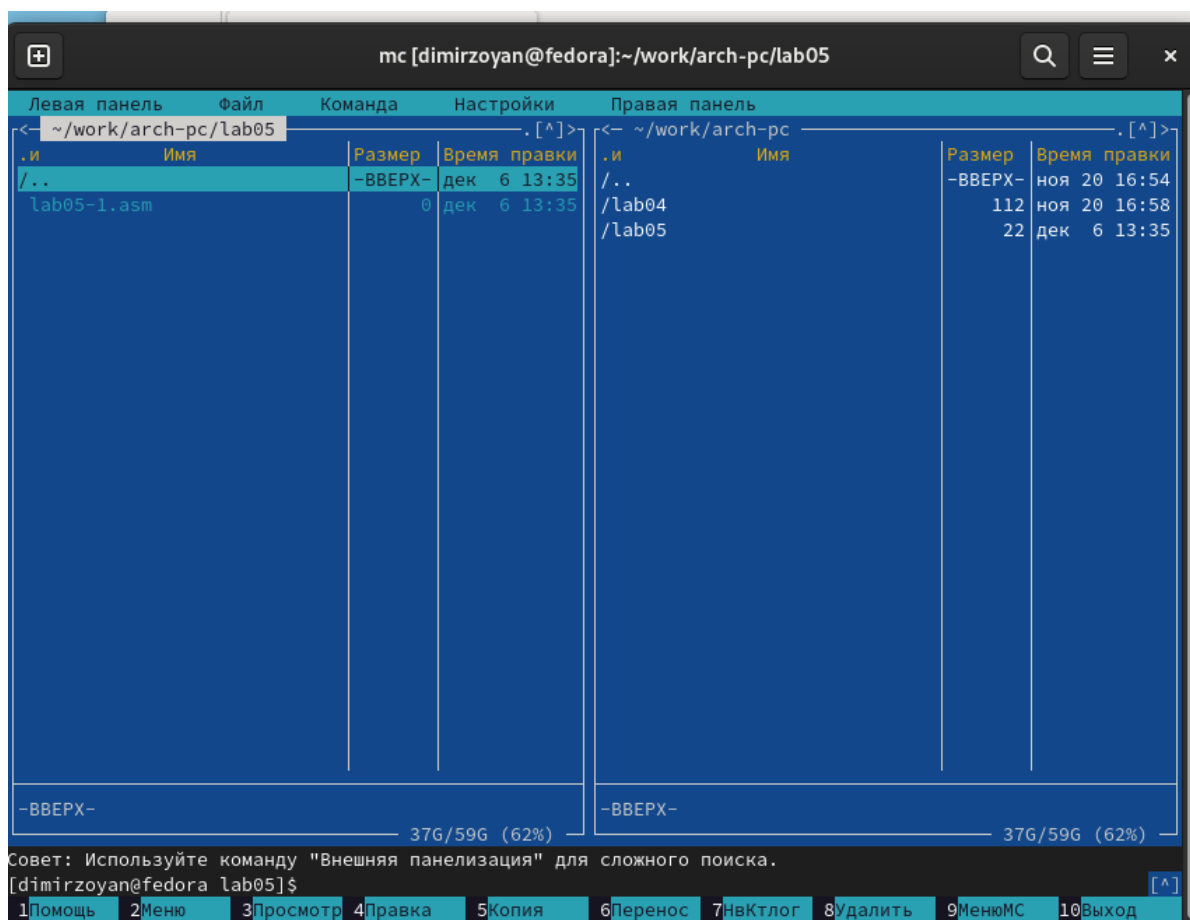
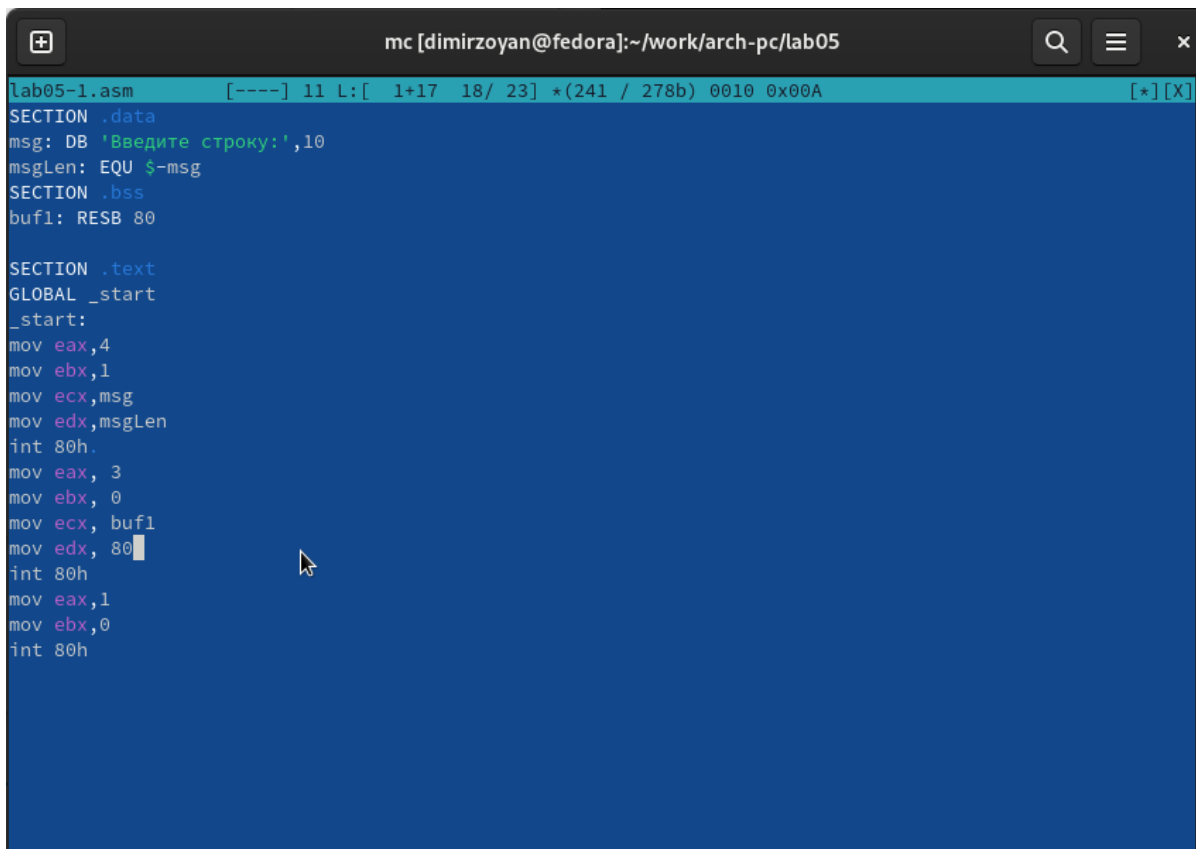


Рис. 2.2: Создание файла lab05-1.asm

Открыл файл на редактирование

Написал код



```
mc [dimirzoyan@fedora]:~/work/arch-pc/lab05
lab05-1.asm [----] 11 L: [ 1+17 18/ 23] *(241 / 278b) 0010 0x00A [*][X]
SECTION .data
msg: DB 'Введите строку:',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. 2.3: Программа в файле lab05-1.asm

Также добавлю код программы в отчет.

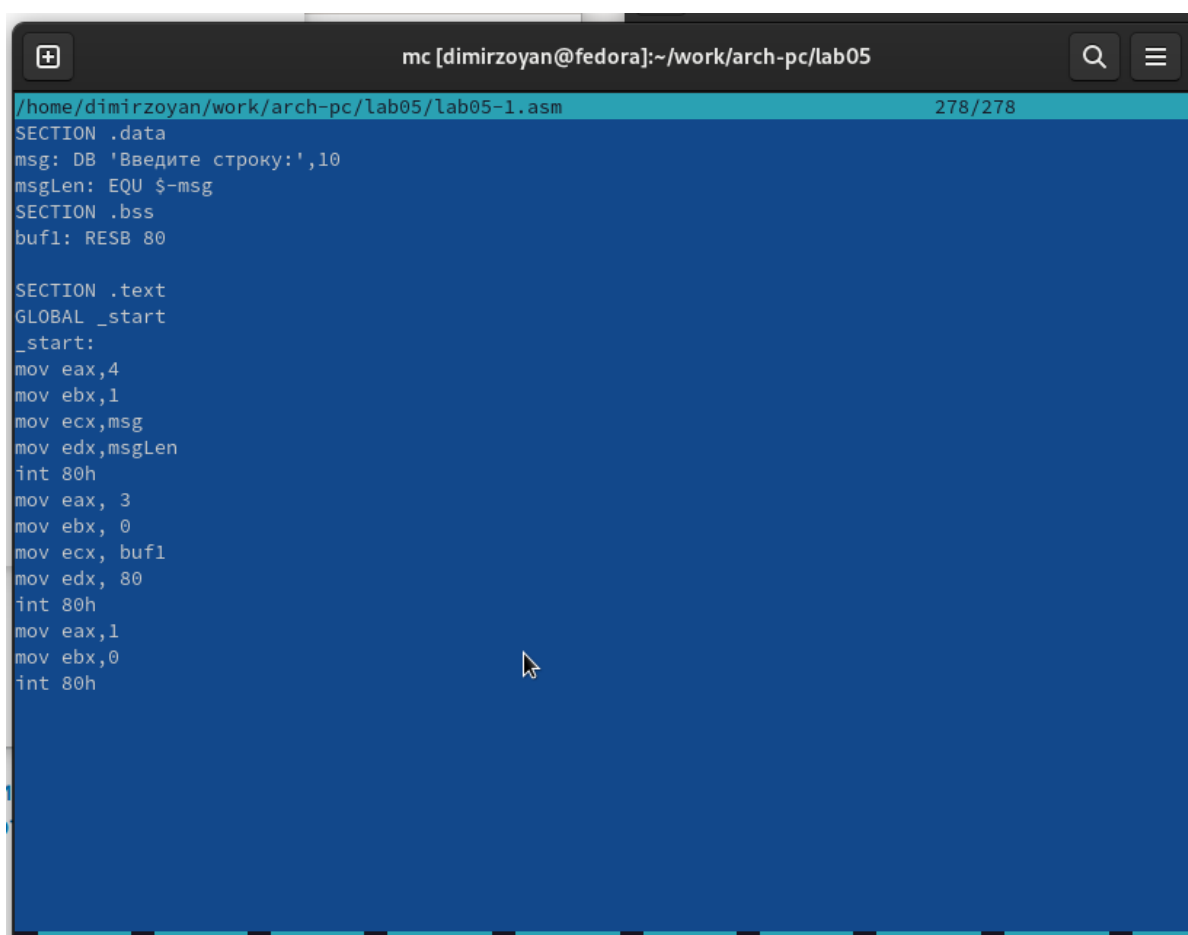
```
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

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
```

Открыл файл на просмотр и убелился, что он содержит набранный код.

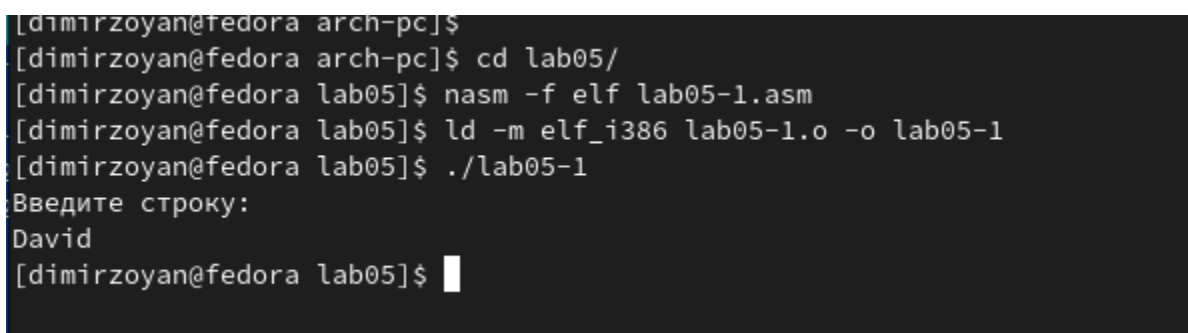


```
mc [dimirzoyan@fedora]:~/work/arch-pc/lab05
/home/dimirzoyan/work/arch-pc/lab05/lab05-1.asm 278/278
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

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. 2.4: Просмотр файла lab05-1.asm

Получил исполняемый файл программы и проверил ее работу.



```
[dimirzoyan@fedora arch-pc]$
[dimirzoyan@fedora arch-pc]$ cd lab05/
[dimirzoyan@fedora lab05]$ nasm -f elf lab05-1.asm
[dimirzoyan@fedora lab05]$ ld -m elf_i386 lab05-1.o -o lab05-1
[dimirzoyan@fedora lab05]$ ./lab05-1
Введите строку:
David
[dimirzoyan@fedora lab05]$
```

Рис. 2.5: Сборка и проверка программы lab05-1.asm

Скачал файл in\_out.asm.

Добавил файл in\_out.asm в рабочий каталог.

Скопировал lab05-1.asm в lab05-2.asm.

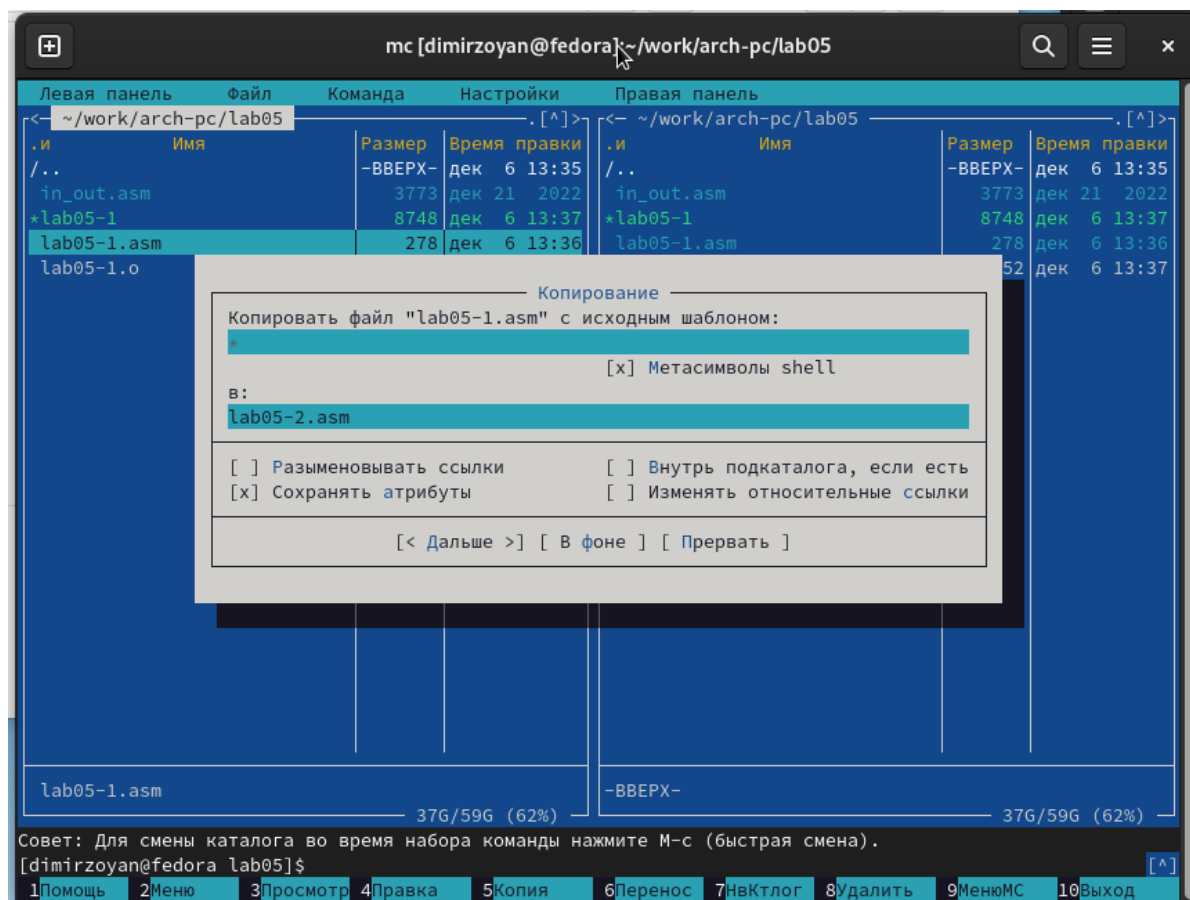
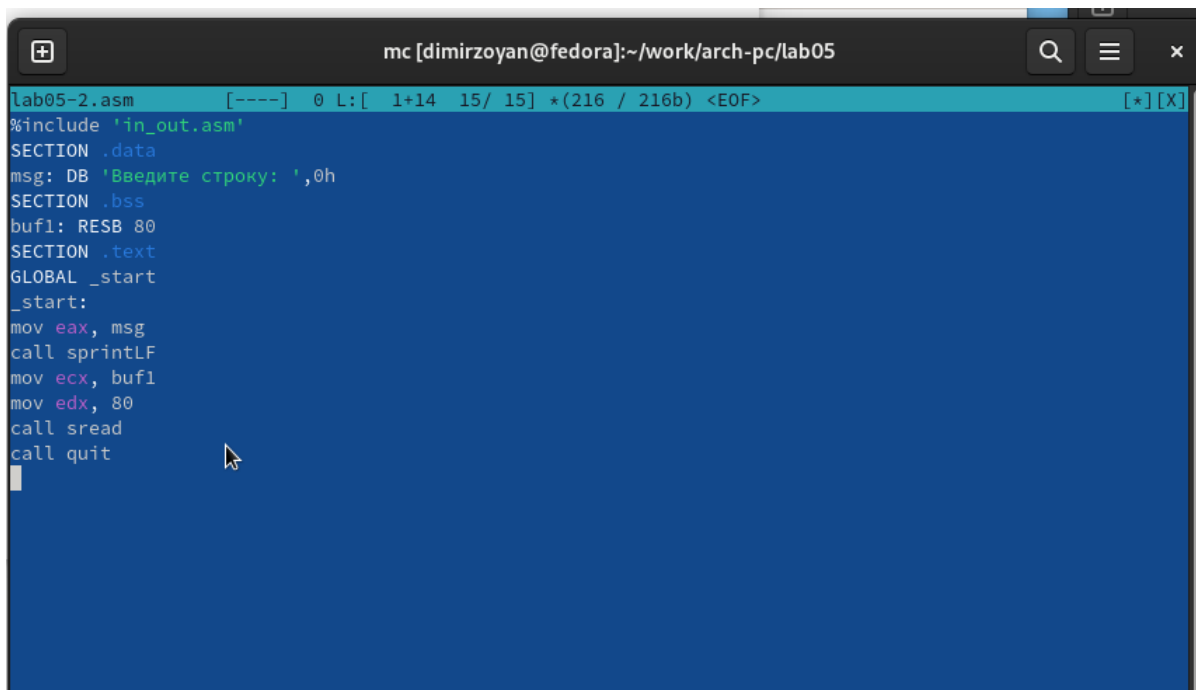


Рис. 2.6: Копирование файла

Написал код программы lab05-2.asm. Скомпилировал программу и проверили запуск.



```
mc [dimirzoyan@fedora]:~/work/arch-pc/lab05
lab05-2.asm [----] 0 L: [ 1+14 15/ 15] *(216 / 216b) <EOF> [*][X]
#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. 2.7: Программа в файле lab05-2.asm

Добавлю код программы в отчет.

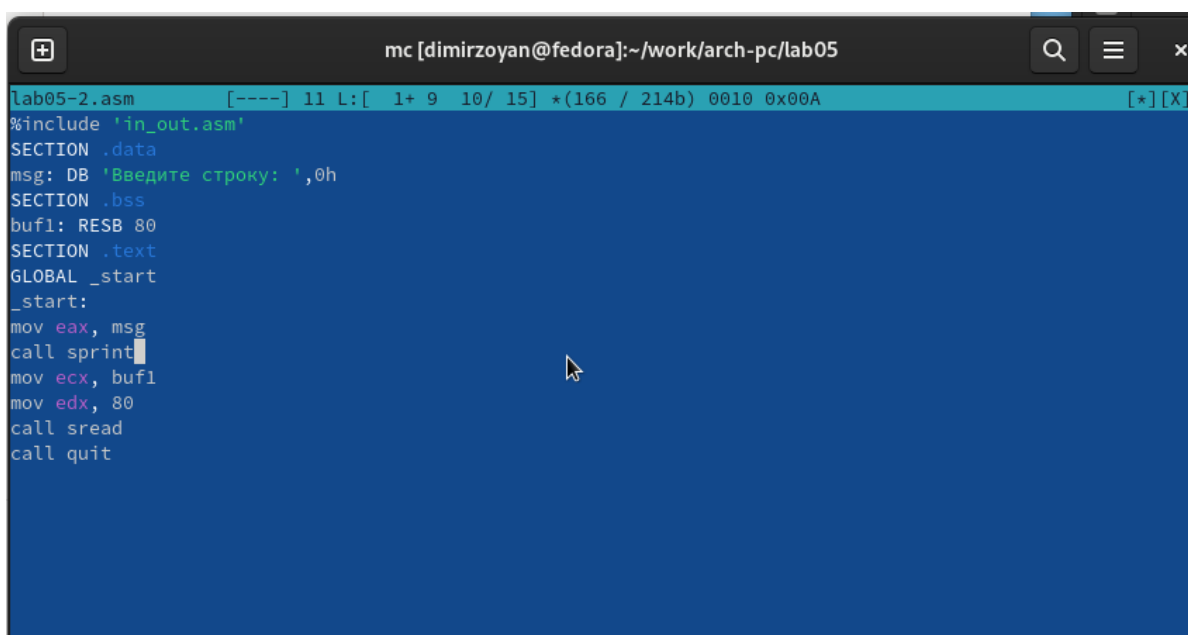
```
%include 'in_out.asm'
SECTION .data
msg: DB 'Введите строку: ',0h
SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:
mov eax, msg
call sprintLF
mov ecx, buf1
mov edx, 80
call sread
```

```
call quit
```

```
[dimirzoyan@fedora lab05]$  
[dimirzoyan@fedora lab05]$ nasm -f elf lab05-2.asm  
[dimirzoyan@fedora lab05]$ ld -m elf_i386 lab05-2.o -o lab05-2  
[dimirzoyan@fedora lab05]$ ./lab05-2  
Введите строку:  
David  
[dimirzoyan@fedora lab05]$
```

Рис. 2.8: Сборка и проверка программы lab05-2.asm

В файле lab5-2.asm заменил подпрограмму sprintLF на sprint. Заново собрал исполняемый файл. Теперь после вывода строки она не завершается символом перехода на новую строку.



```
mc [dimirzoyan@fedora]:~/work/arch-pc/lab05  
lab05-2.asm [----] 11 L: [ 1+ 9 10/ 15] *(166 / 214b) 0010 0x00A [*][X]  
%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. 2.9: Программа в файле lab05-2.asm

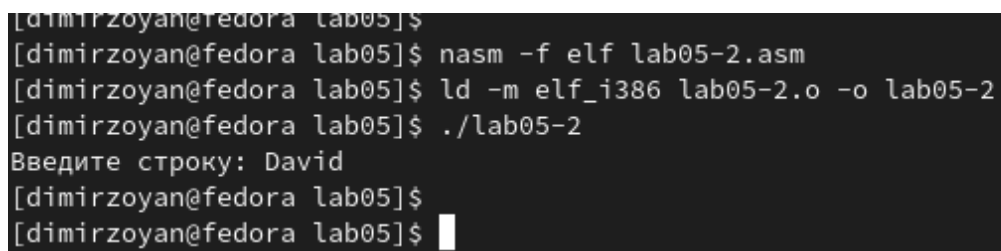
Покажу код программы в отчете.

```
%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

```



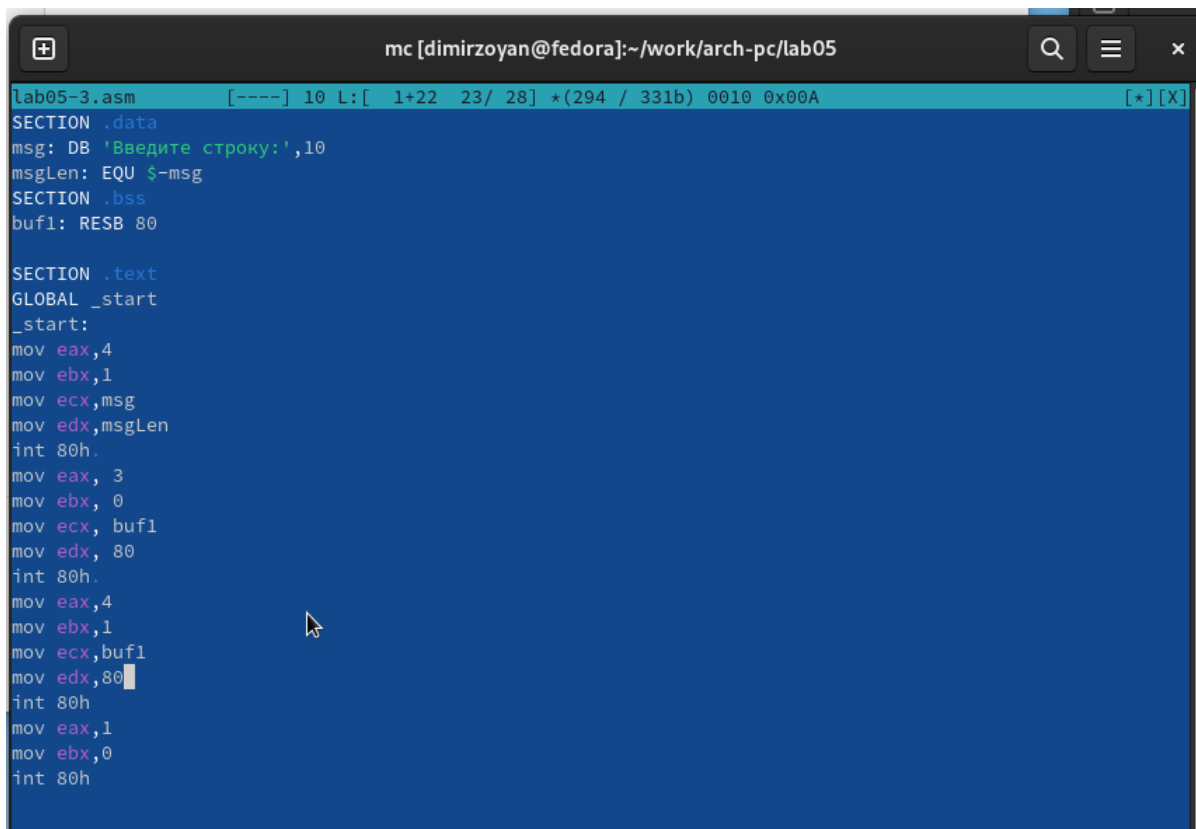
```

[dimirzoyan@fedora lab05]$
[dimirzoyan@fedora lab05]$ nasm -f elf lab05-2.asm
[dimirzoyan@fedora lab05]$ ld -m elf_i386 lab05-2.o -o lab05-2
[dimirzoyan@fedora lab05]$ ./lab05-2
Введите строку: David
[dimirzoyan@fedora lab05]$
[dimirzoyan@fedora lab05]$ █

```

Рис. 2.10: Сборка и проверка программы lab05-2.asm

Скопировал программу lab05-1.asm и изменил код, чтобы вывести приглашение типа “Введите строку:”, ввести строку с клавиатуры, вывести введённую строку на экран.



```
mc [dimirzoyan@fedora]:~/work/arch-pc/lab05
lab05-3.asm [----] 10 L: [ 1+22 23/ 28] *(294 / 331b) 0010 0x00A [*][X]
SECTION .data
msg: DB 'Введите строку:',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,80
int 80h
mov eax,1
mov ebx,0
int 80h
```

Рис. 2.11: Программа в файле lab05-3.asm

Покажу код программы в отчете.

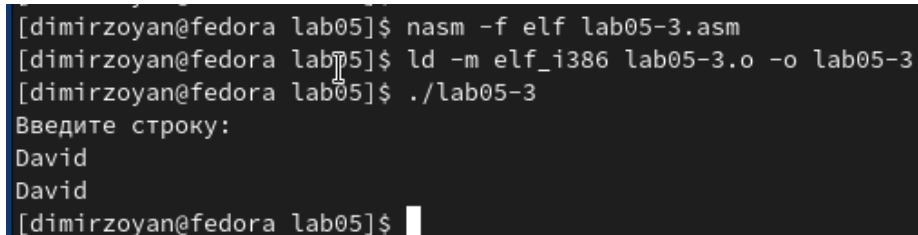
```
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

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

```



```

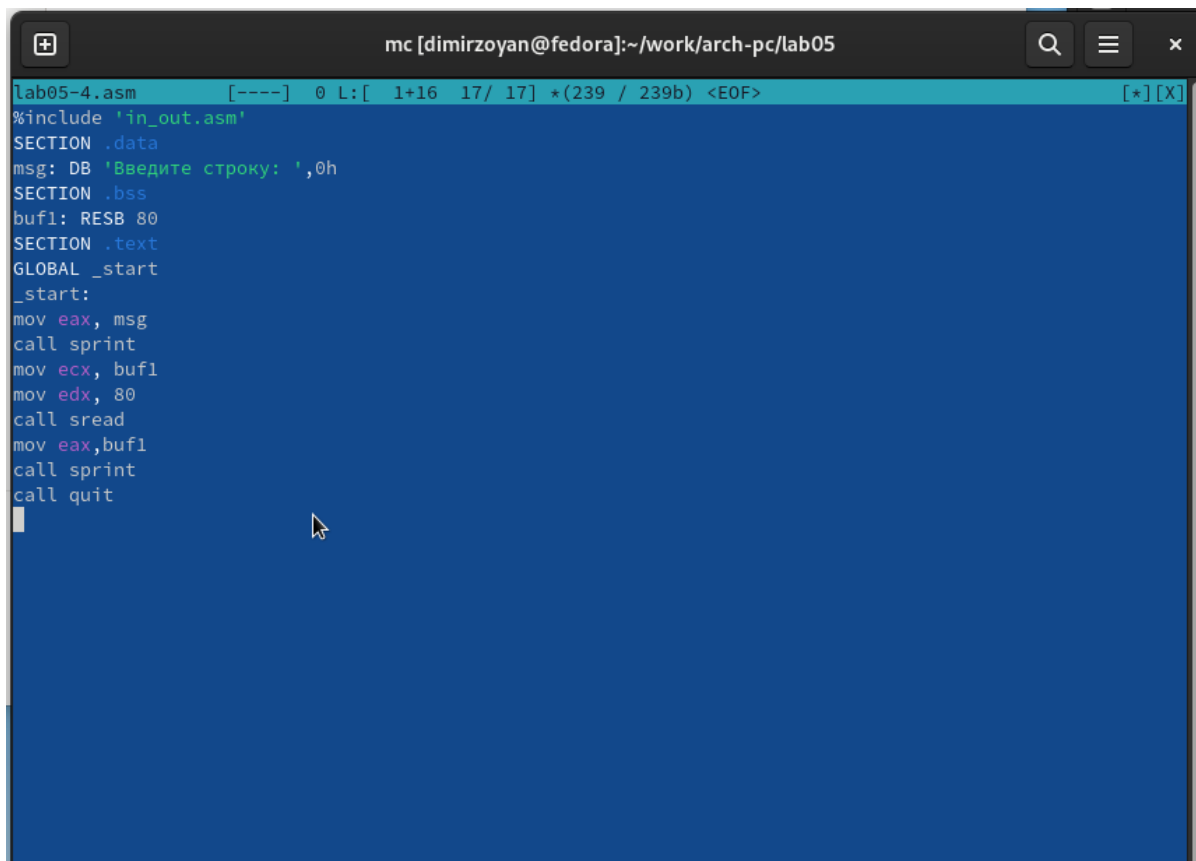
[dimirzoyan@fedora lab05]$ nasm -f elf lab05-3.asm
[dimirzoyan@fedora lab05]$ ld -m elf_i386 lab05-3.o -o lab05-3
[dimirzoyan@fedora lab05]$ ./lab05-3
Введите строку:
David
David
[dimirzoyan@fedora lab05]$

```

Рис. 2.12: Сборка и проверка программы lab05-3.asm

Скопировал программу lab05-2.asm и изменил код, чтобы вывести приглашение типа “Введите строку:”, ввести строку с клавиатуры, вывести введённую строку на экран.





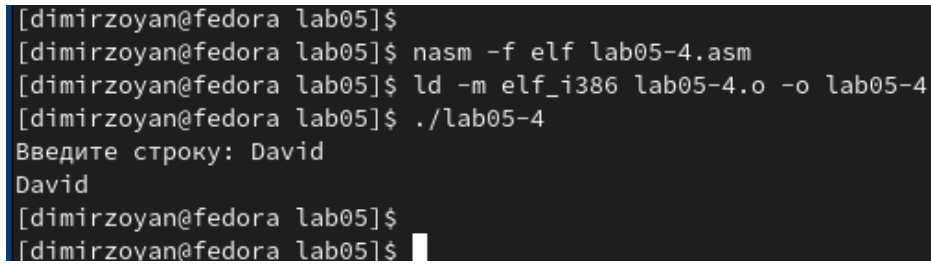
```
lab05-4.asm      [----]  0 L: [ 1+16 17/ 17] *(239 / 239b) <EOF>  [*] [X]
#include 'in_out.asm'
SECTION .data
msg: DB 'Введите строку: ',0h
SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:
mov eax, msg
call sprint
mov ecx, buf1
mov edx, 80
call sread
mov eax, buf1
call sprint
call quit
```

Рис. 2.13: Программа в файле lab05-4.asm

Покажу код программы в отчете.

```
%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 sprint
call quit
```

A terminal window with a dark background and light-colored text. The prompt is [dimirzoyan@fedora lab05]\$. The user enters 'nasm -f elf lab05-4.asm', then 'ld -m elf\_i386 lab05-4.o -o lab05-4', and finally './lab05-4'. The program prompts 'Введите строку: David' and the user enters 'David'. The terminal then shows the prompt again, and the user enters a space character.

```
[dimirzoyan@fedora lab05]$
[dimirzoyan@fedora lab05]$ nasm -f elf lab05-4.asm
[dimirzoyan@fedora lab05]$ ld -m elf_i386 lab05-4.o -o lab05-4
[dimirzoyan@fedora lab05]$ ./lab05-4
Введите строку: David
David
[dimirzoyan@fedora lab05]$
[dimirzoyan@fedora lab05]$ 
```

Рис. 2.14: Сборка и проверка программы lab05-4.asm

Отличие этих двух реализаций в том, что файл `in_out.asm` содержит уже готовые подпрограммы для обеспечения ввода/вывода. Таким образом, нам остается только разместить данные в нужных регистрах и вызвать желаемую подпрограмму с помощью `call`.

## 3 Выводы

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