

```
--  
## Front matter  
title: "Отчет по лабораторной работе №5"  
subtitle: "Дисциплина: архитектура компьютера"  
author: "Мазурский Александр Дмитриевич"  
  
## Generic options  
lang: ru-RU  
toc-title: "Содержание"  
  
## Bibliography  
bibliography: bib/cite.bib  
csl: pandoc/csl/gost-r-7-0-5-2008-numeric.csl  
  
## Pdf output format  
toc: true # Table of contents  
toc-depth: 2  
lof: true # List of figures  
lot: true # List of tables  
fontsize: 12pt  
linestretch: 1.5  
papersize: a4  
documentclass: scrreprt  
## I18n polyglossia  
polyglossia-lang:  
  name: russian  
  options:  
    - spelling=modern  
    - babelshorthands=true  
polyglossia-otherlangs:  
  name: english
```

```
## I18n babel
babel-lang: russian
babel-otherlangs: english
## Fonts
mainfont: IBM Plex Serif
romanfont: IBM Plex Serif
sansfont: IBM Plex Sans
monofont: IBM Plex Mono
mathfont: STIX Two Math
mainfontoptions: Ligatures=Common,Ligatures=TeX,Scale=0.94
romanfontoptions: Ligatures=Common,Ligatures=TeX,Scale=0.94
sansfontoptions: Ligatures=Common,Ligatures=TeX,Scale=MatchLowercase,Scale=0.94
monofontoptions: Scale=MatchLowercase,Scale=0.94,FakeStretch=0.9
mathfontoptions:
## Biblatex
biblatex: true
biblio-style: "gost-numeric"
biblatexoptions:
- parentracker=true
- backend=biber
- hyperref=auto
- language=auto
- autolang=other*
- citestyle=gost-numeric
## Pandoc-crossref LaTeX customization
figureTitle: "Рис."
tableTitle: "Таблица"
listingTitle: "Листинг"
lofTitle: "Список иллюстраций"
lotTitle: "Список таблиц"
lolTitle: "Листинги"
## Misc options
```

```
indent: true

header-includes:
- \usepackage{indentfirst}
- \usepackage{float} # keep figures where there are in the text
- \floatplacement{figure}{H} # keep figures where there are in the text

---
```

Цель работы

Целью данной лабораторной работы является приобретение практических навыков работы в Midnight Commander, освоение инструкций языка ассемблера mov и int.

Задание

1. Основы работы с mc
2. Структура программы на языке ассемблера NASM
3. Подключение внешнего файла
4. Выполнение заданий для самостоятельной работы

Теоретическое введение

Midnight Commander (или просто mc) — это программа, которая позволяет просматривать структуру каталогов и выполнять основные операции по управлению файловой системой, т.е. mc является файловым менеджером. Midnight Commander позволяет сделать работу с файлами более удобной и наглядной. Программа на языке ассемблера NASM, как правило, состоит из трёх секций: секция кода программы (SECTION .text), секция инициализированных (известных во время компиляции) данных (SECTION .data) и секция неинициализированных данных (тех, под которые во время компиляции только отводится память, а значение присваивается в ходе выполнения программы) (SECTION .bss). Для объявления инициализированных данных в секции .data используются директивы DB, DW, DD, DQ и DT, которые резервируют память и указывают, какие значения должны храниться в этой памяти:

- DB (define byte) — определяет переменную размером в 1 байт;
- DW (define word) — определяет переменную размером в 2 байта (слово);
- DD (define double word) — определяет переменную размером в 4 байта (двойное слово);
- DQ (define quad word) — определяет переменную размером в 8 байт (четвернное слово);
- DT (define ten bytes) — определяет переменную размером в 10 байт. Директивы используются для объявления простых переменных и для объявления массивов. Для определения строк принято использовать директиву DB в связи с особенностями хранения данных в оперативной памяти. Инструкция языка ассемблера mov предназначена для дублирования данных источника в приёмнике.

```
mov dst, src
```

Здесь operand dst — приёмник, a src — источник. В качестве operandов могут выступать регистры (register), ячейки памяти (memory) и непосредственные значения (const). Инструкция языка ассемблера int предназначена для вызова прерывания с указанным номером.

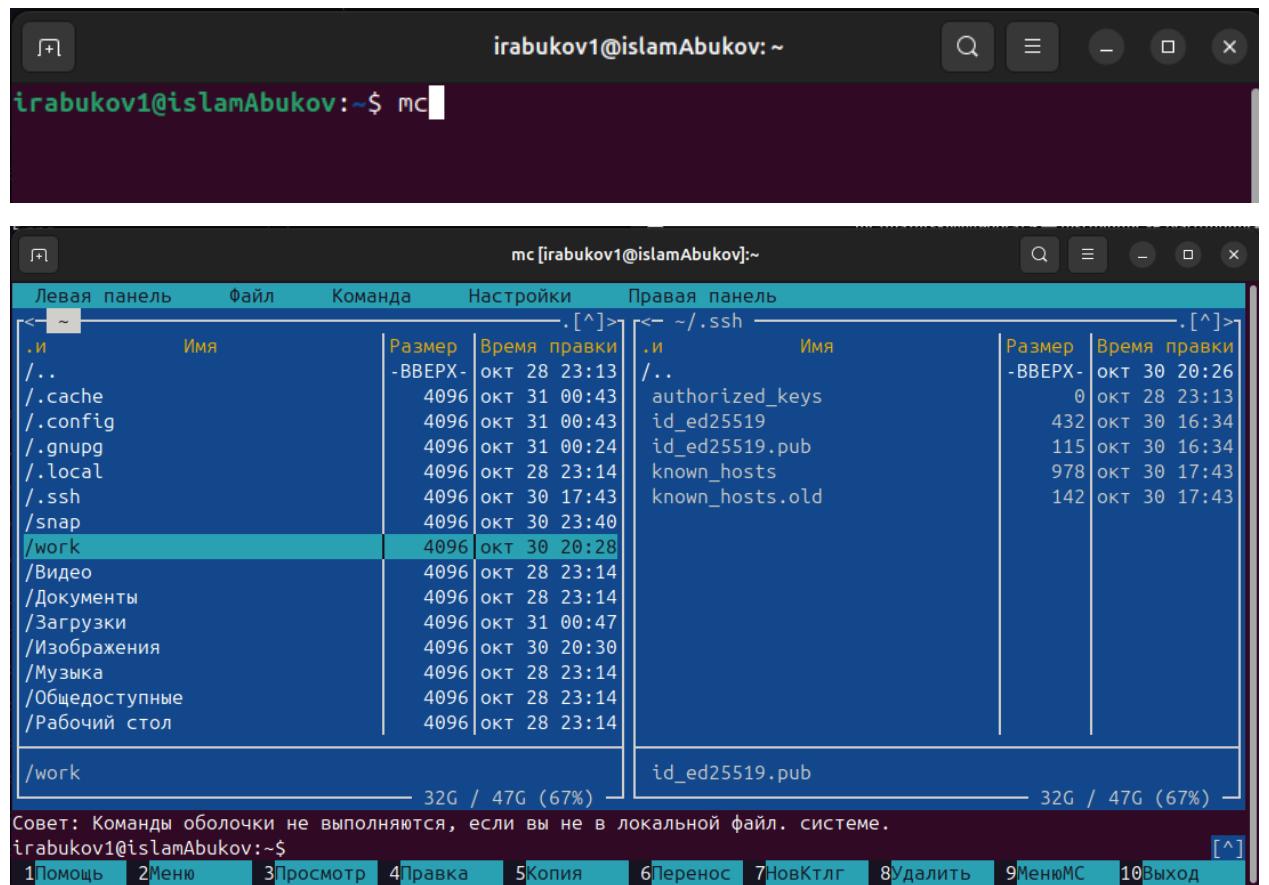
```
int n
```

Здесь n — номер прерывания, принадлежащий диапазону 0–255. При программировании в Linux с использованием вызовов ядра sys_calls n=80h (принято задавать в шестнадцатеричной системе счисления).

Выполнение лабораторной работы

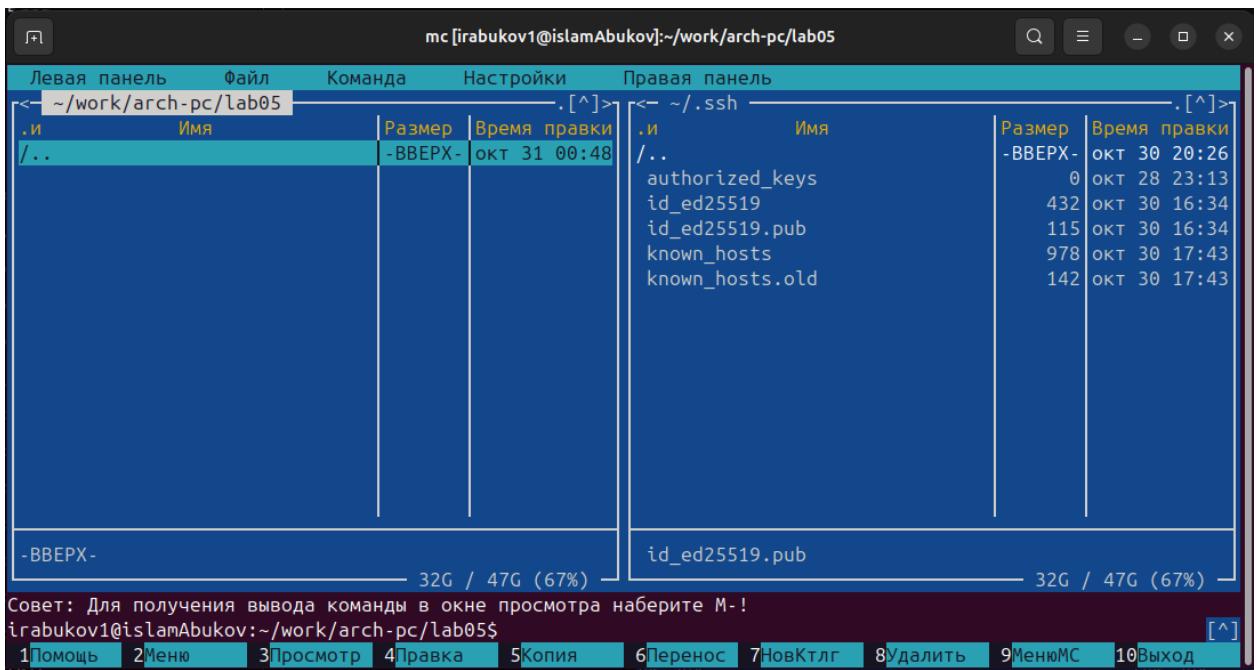
Основы работы с Midnight Commander

Введя соответствующую команду в терминале, я открываю Midnight Commander

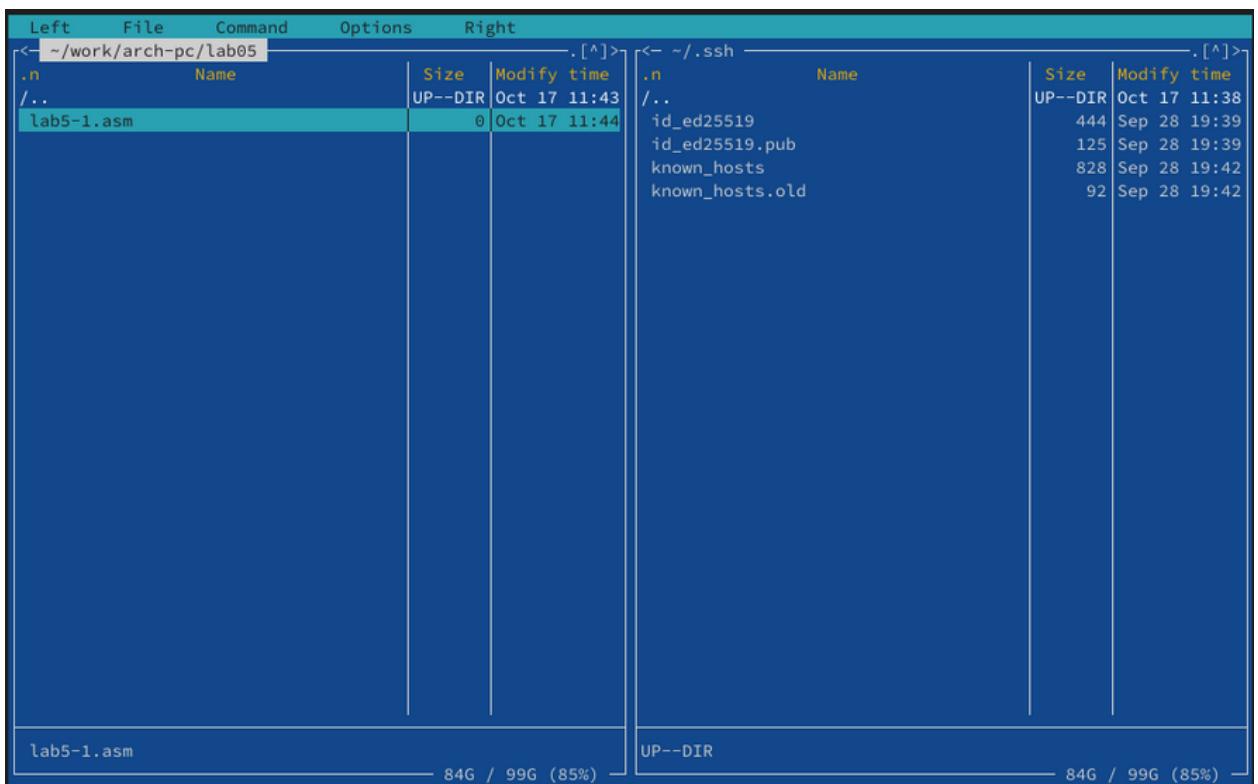


Перехожу в созданный каталог в предыдущей лабораторной работе

С помощью функциональной клавиши, я создаю подкаталог lab05, в котором буду работать



В строке ввода ввожу команду touch и создаю файл



Работа в NASM

С помощью F4 открываю только что созданный файл и вношу код с листинга

```
GNU nano 7.2                               mc [irabukov1@islamAbukov]:~/work/arch-pc/lab05
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
```

Справка Выход Записать ЧитФайл Поиск Замена Вырезать Вставить Выполнить Выровнять Позиция К строке Отмена Повтор

Проверяю сохраненные изменения с помощью клавиши F3

```
GNU nano 7.2                               mc [irabukov1@islamAbukov]:~/work/arch-pc/lab05
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
```

[Записано 26 строк]
Справка Выход Записать ЧитФайл Поиск Замена Вырезать Вставить Выполнить Выровнять Позиция К строке Отмена Повтор

Транслирую и компоную измененный файл, запускаю

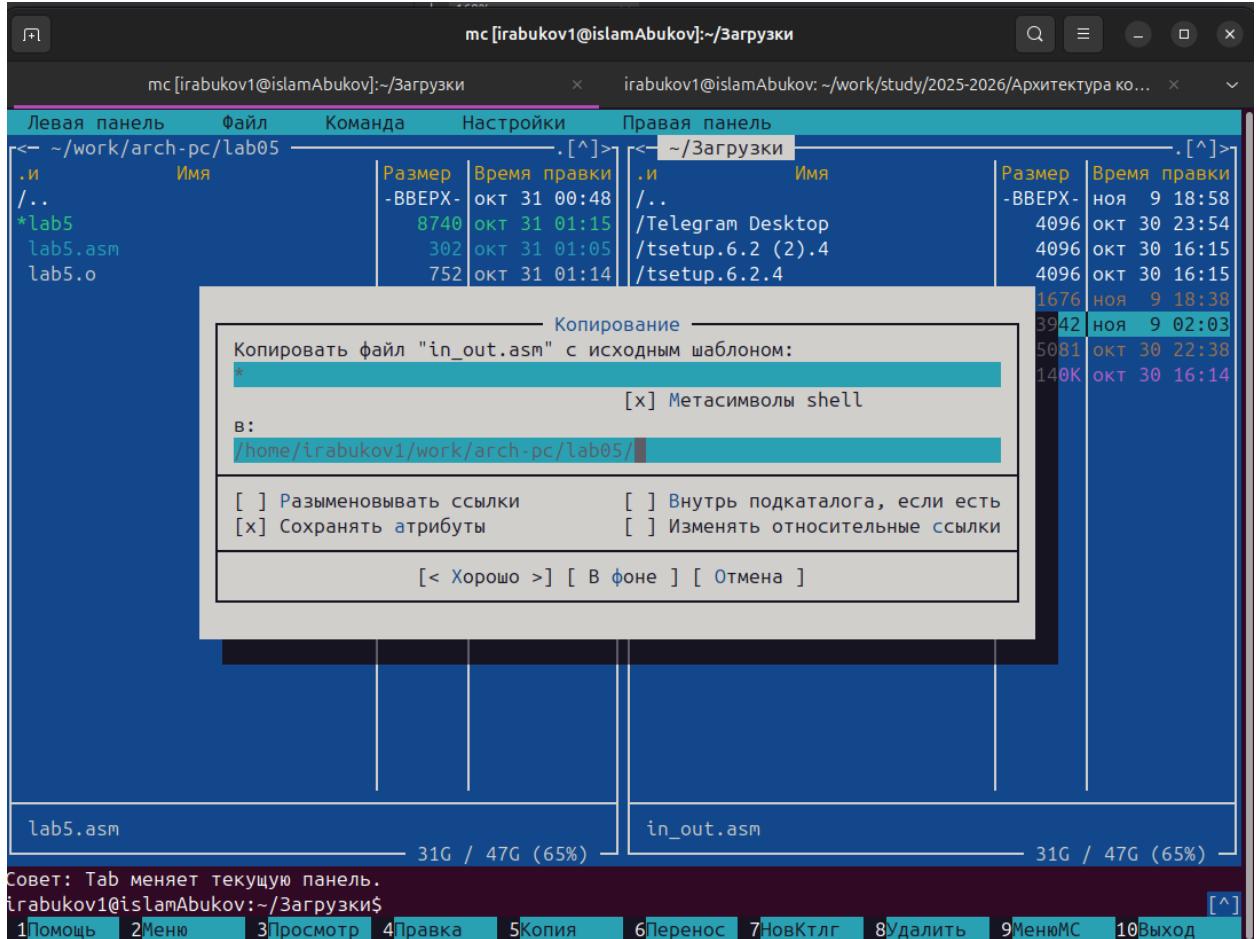
```

irabukov1@islamAbukov:~$ cd ~/work/arch-pc/lab05
irabukov1@islamAbukov:~/work/arch-pc/lab05$ nasm -f elf lab5.asm
irabukov1@islamAbukov:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5 lab5.o
irabukov1@islamAbukov:~/work/arch-pc/lab05$ ./lab5
Введите строку
Абуков Ислам Ренатович
irabukov1@islamAbukov:~/work/arch-pc/lab05$ 

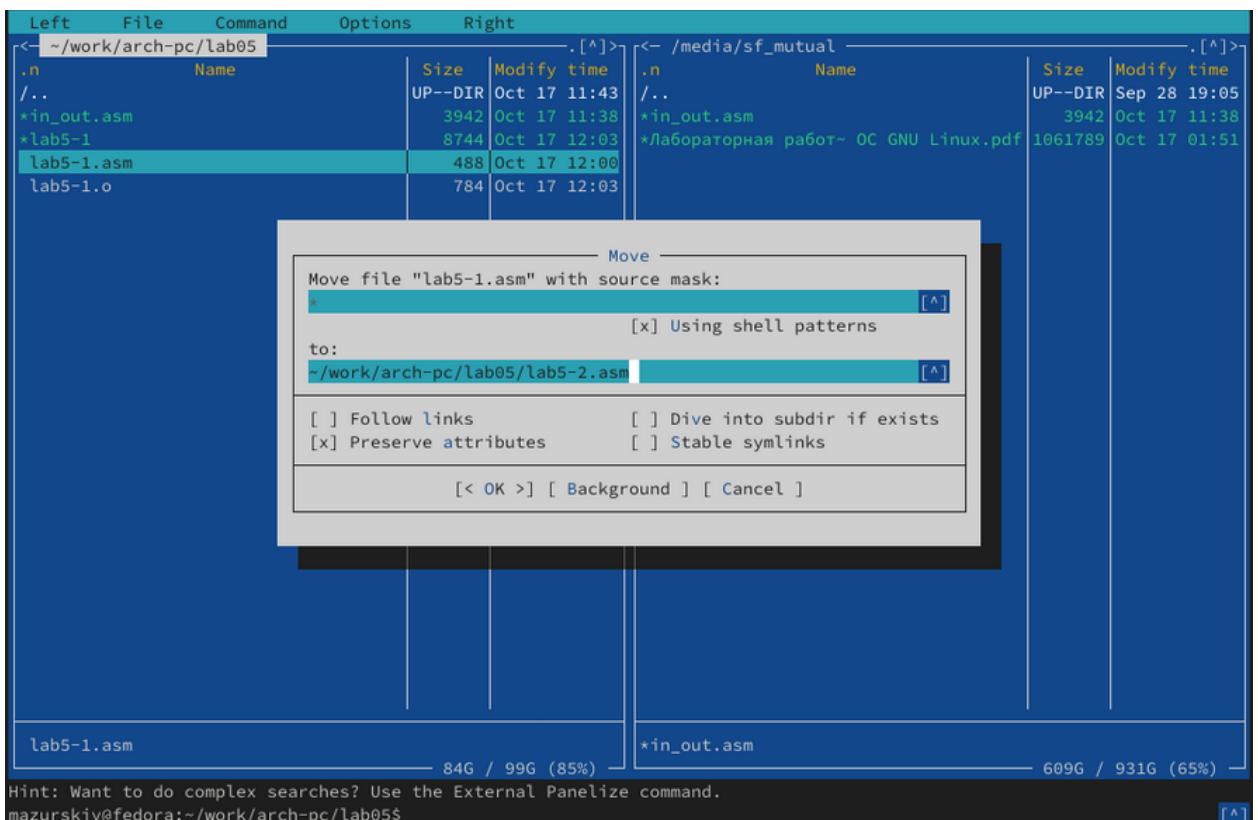
```

Подключение внешнего файла

Скачанный с ТУИС файл сохраняю в общую папку на своем компьютере, на виртуальной машине в интерфейсе Midnight Commander перехожу в директорию общей папки, копирую файл в рабочий подкаталог.



Создаю копию файла для последующей работы с ним



В копии файла подключаю подпрограмм из подключенного файла

```
mc [irabukov1@islamAbukov]:~/work/arch-pc/lab05
mc [irabukov1@islamAbukov]:~/work/arch-pc/lab05      irabukov1@islamAbukov: ~/work/study/2025-2026/Архитектура ко...
GNU nano 7.2          /home/irabukov1/work/arch-pc/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
    call   quit
```

[Прочитана 21 строка]

Справка **Записать** **Поиск** **Вырезать** **Выполнить** **Позиция** **Отмена**
Выход **ЧитФайл** **Замена** **Вставить** **Выровнять** **К строке** **Повтор**

Транслирую, компоную и запускаю программу с подключенным файлом

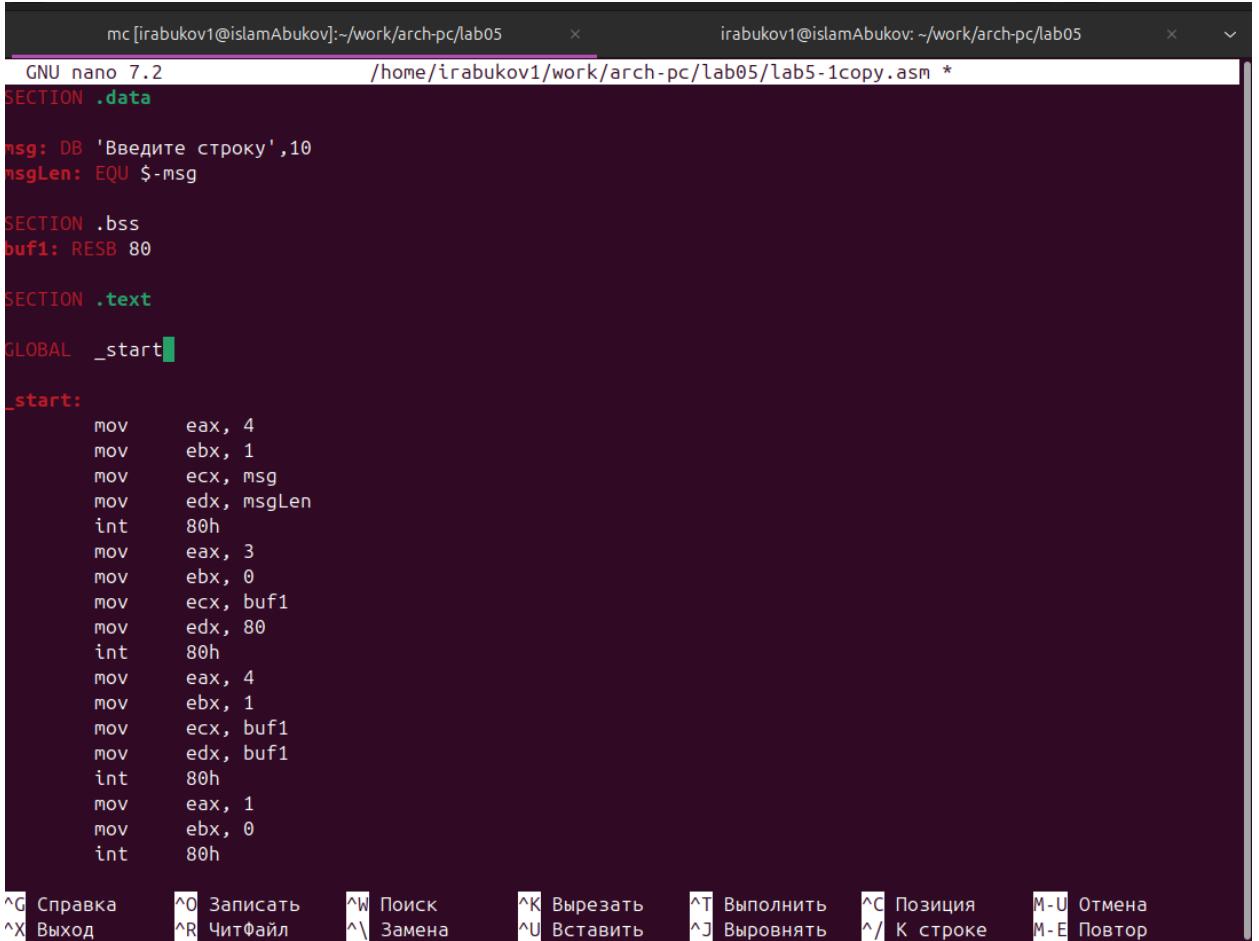
```
irabukov1@islamAbukov:~/work/arch-pc/lab05$ nasm -f elf lab5-2.asm
irabukov1@islamAbukov:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-22 lab5-2.o
irabukov1@islamAbukov:~/work/arch-pc/lab05$ ./lab5-22
Введите строку: Islam Abukov
irabukov1@islamAbukov:~/work/arch-pc/lab05$
```

Редактирую файл и заменяю в нем подпрограмму sprintLF на sprint. Разница подпрограмм в том, что вторая вызывает ввод на той же строке.

```
irabukov1@islamAbukov:~/work/arch-pc/lab05$ nasm -f elf lab5-2.asm
irabukov1@islamAbukov:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-22 lab5-2.o
irabukov1@islamAbukov:~/work/arch-pc/lab05$ ./lab5-22
Введите строку: Islam Abukov
irabukov1@islamAbukov:~/work/arch-pc/lab05$
```

Задание для самостоятельной работы

Создаю копию lab5-1.asm, редактирую так, чтобы в конце выводилась введенная мною строка с клавиатуры



```
mc [irabukov1@islamAbukov]:~/work/arch-pc/lab05          irabukov1@islamAbukov:~/work/arch-pc/lab05
GNU nano 7.2                                              /home/irabukov1/work/arch-pc/lab05/lab5-1copy.asm *
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, buf1
    int    80h
    mov    eax, 1
    mov    ebx, 0
    int    80h

^G Справка      ^O Записать      ^W Поиск      ^K Вырезать      ^T Выполнить      ^C Позиция      M-U Отмена
^X Выход      ^R Читфайл      ^\ Замена      ^U Вставить      ^J Выровнять      ^/ К строке      M-E Повтор
```

Транслирую, компоную и запускаю свою программу

```
irabukov1@islamAbukov:~/work/arch-pc/lab05$ nasm -f elf lab5-1copy.asm
irabukov1@islamAbukov:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1copy lab5-1copy.o
irabukov1@islamAbukov:~/work/arch-pc/lab05$ ./lab5-1copy
Введите строку
sdgfsdgfsf
sdgfsdgfsf
irabukov1@islamAbukov:~/work/arch-pc/lab05$
```

Код прикладываю

```
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, buf1
    int    80h
    mov    eax, 1
    mov    ebx, 0
    int    80h
```

Создаю копию lab5-2.asm, редактирую так, чтобы в конце выводилась введеная мною строка с клавиатуры

```

GNU nano 7.2                               /home/irabukov1/work/arch-pc/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

```

[Прочитано 28 строк]
 ^G Справка ^O Записать ^W Поиск ^K Вырезать ^T Выполнить ^C Позиция M-U Отмена
 ^X Выход ^R ЧитФайл ^\| Замена ^U Вставить ^J Выровнять ^/ К строке M-E Повтор

Транслирую, компоную и запускаю свою программу

```

irabukov1@islamAbukov:~/work/arch-pc/lab05$ nasm -f elf lab5-2copy.asm
irabukov1@islamAbukov:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-2copy lab5-2copy.o
irabukov1@islamAbukov:~/work/arch-pc/lab05$ ./lab5-2copy
Введите строку: Абуков Ислам
Абуков Ислам
irabukov1@islamAbukov:~/work/arch-pc/lab05$
```

Код прикладываю

%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

Выводы

При выполнении данной лабораторной работы я приобрёл практические навыки работы в *Midnight Commander*, а также освоил инструкции языка ассемблера *mov* и *int*.