

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Компьютерные и информационные науки

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ № 5

дисциплина: Архитектура компьютера

Студент: Нечаева Кира

Группа: НКАбд-04-23

МОСКВА

2023__ г.

Содержание

1. Цель работы.....	3
2. Задание.....	4
3. Выполнение лабораторной работы.....	5
4. Выводы.....	15
5. Источники.....	16

1 Цель работы

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

2 Задание

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

3 Выюлнение лабораторной работы

1. Основы работы с mc

Для начала я открываю Midnight Commander в терминале. (рис. 1)

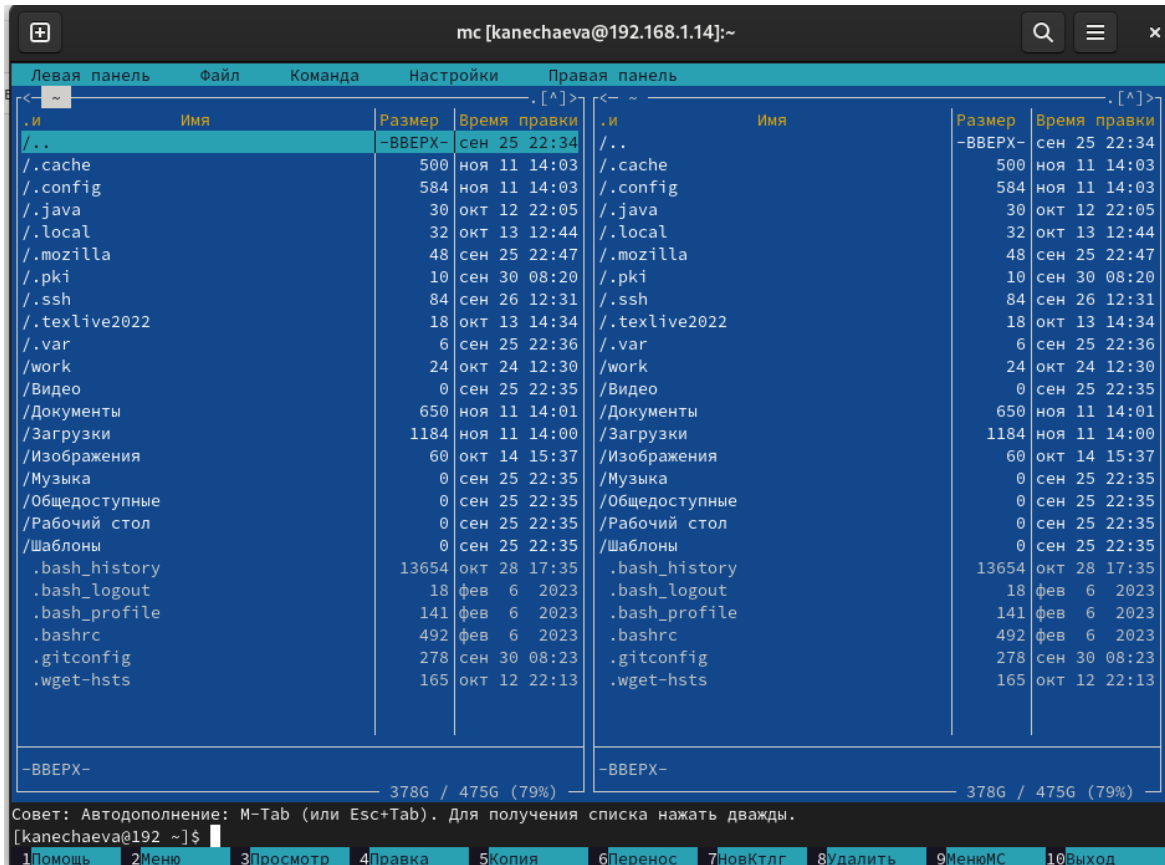


Рис. 1. Окно Midnight Commander

Затем, пользуясь клавишами ↑, ↓ и Enter перехожу в каталог ~/work/arch-pc созданный при выполнении лабораторной работы №4 и с помощью функциональной клавиши F7 создаю папку lab05. (рис. 2)

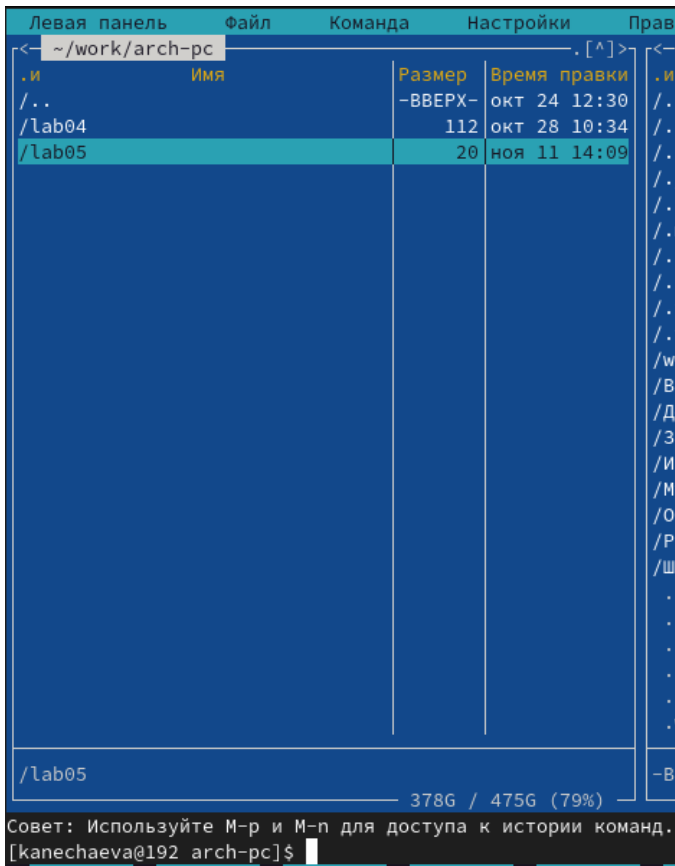


Рис. 2. Создание каталога lab05

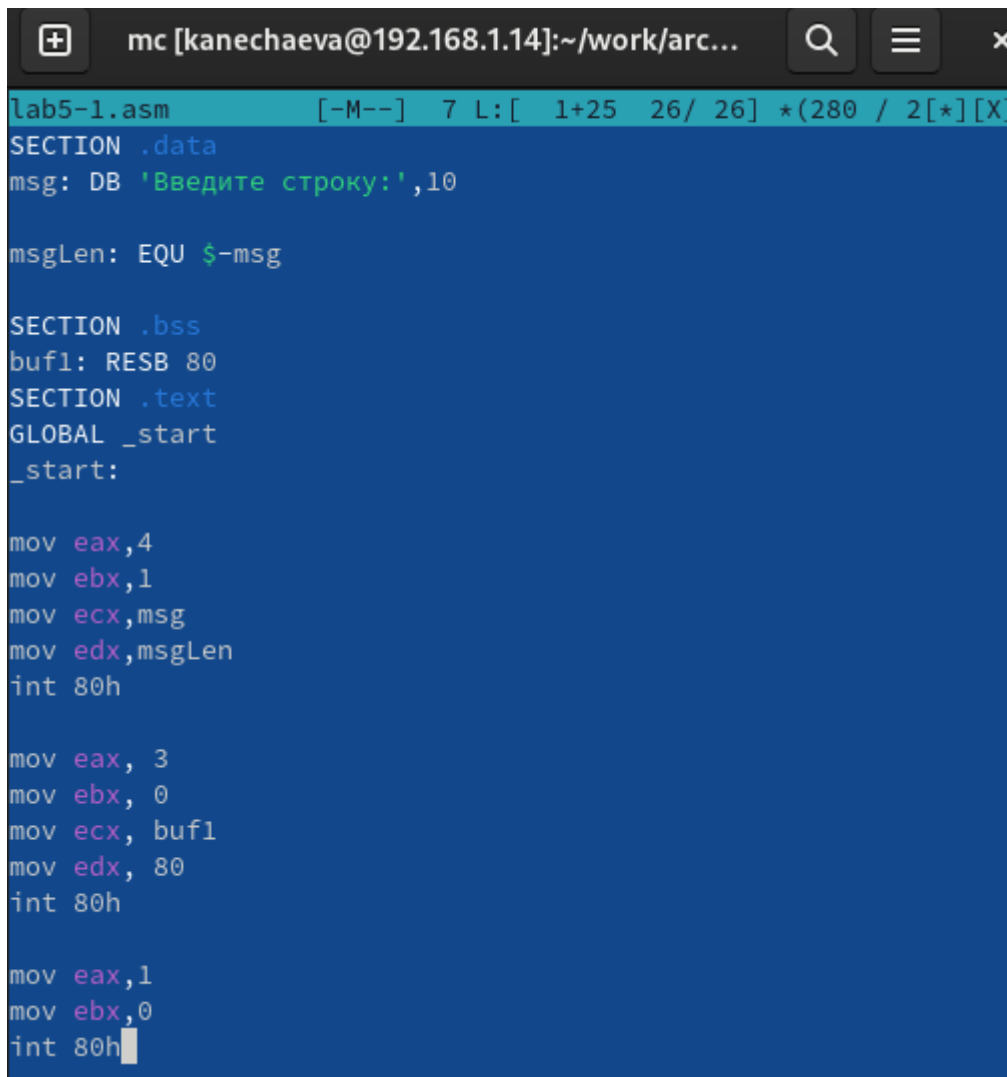
Пользуясь строкой ввода и командой touch создаю файл lab5-1.asm. (рис. 3)



Рис. 3. Созданный файл lab5-1.asm

2. Структура программы на языке ассемблера NASM

С помощью функциональной клавиши F4 открываю файл lab5-1.asm для редактирования во встроенном редакторе. Затем ввожу текст программы из листинга 5.1 без комментариев, после чего сохраняю изменения и закрываю файл. (рис. 4)



```
lab5-1.asm [-M--] 7 L: [ 1+25 26/ 26] *(280 / 2[*])[X]
SECTION .data
msg: DB 'Введите строку:',10

msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:

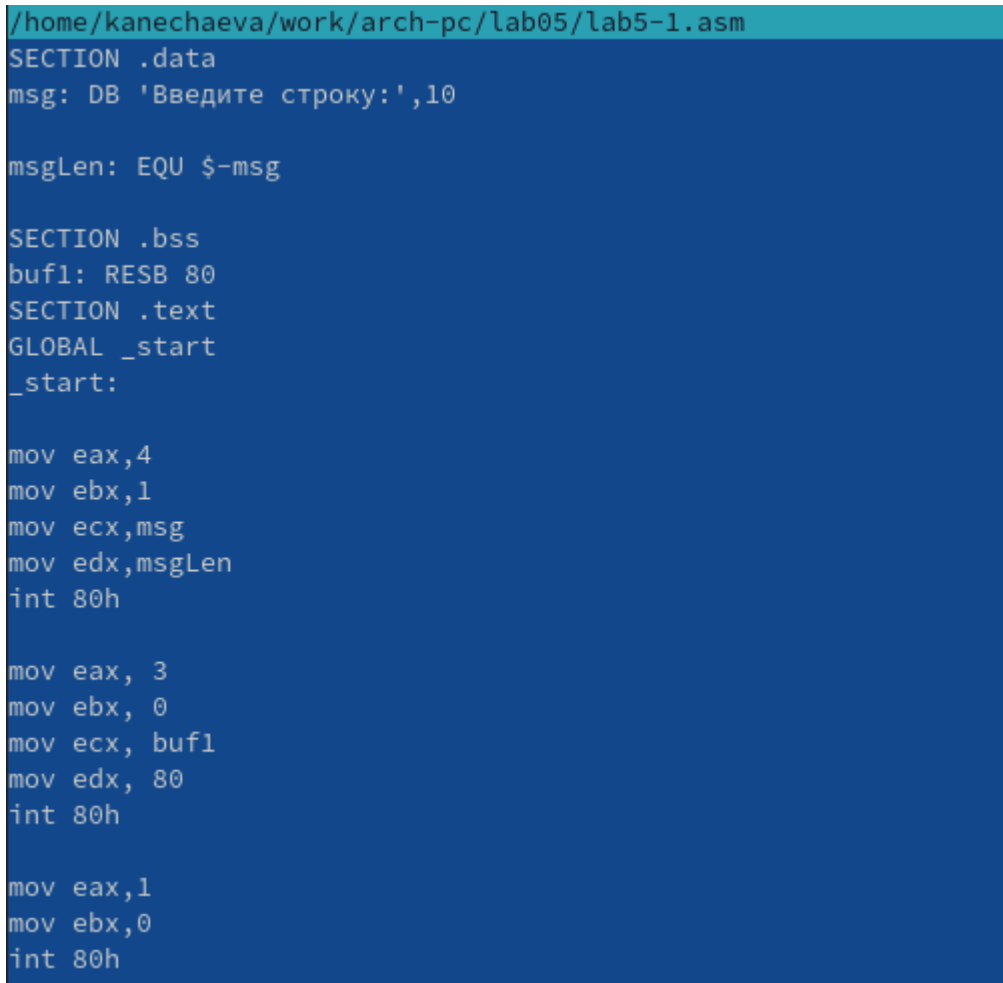
mov eax,4
mov ebx,1
mov ecx,msg
mov edx,msgLen
int 80h

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

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

Рис. 4. Редактирование файла

С помощью клавиши F3 открываю файл lab5-1.asm для просмотра. Убеждаюсь, что файл содержит текст программы. (рис. 5)



```
/home/kanechaeva/work/arch-pc/lab05/lab5-1.asm
SECTION .data
msg: DB 'Введите строку:',10

msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:

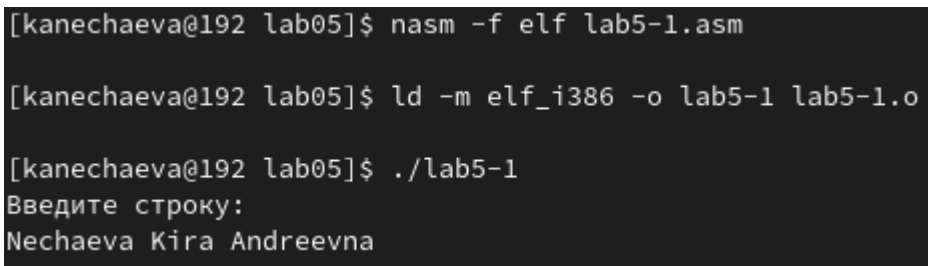
mov eax,4
mov ebx,1
mov ecx,msg
mov edx,msgLen
int 80h

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

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

Рис. 5. Открытие файла для просмотра

Транслирую текст программы lab5-1.asm в объектный файл. Выполняю компоновку объектного файла и запускаю получившийся исполняемый файл. Программа выводит строку 'Введите строку:' и ожидает ввода с клавиатуры. На запрос ввожу свои ФИО. (рис. 6)



```
[kanechaeva@192 lab05]$ nasm -f elf lab5-1.asm

[kanechaeva@192 lab05]$ ld -m elf_i386 -o lab5-1 lab5-1.o

[kanechaeva@192 lab05]$ ./lab5-1
Введите строку:
Mechaeva Kira Andreevna
```

Рис. 6. Компиляция и исполнение файла

1. Подключение внешнего файла in_out.asm

Скачиваю файл in_out.asm со страницы курса в ТУИС и перемещаю его в недавно созданный каталог lab05.

С помощью функциональной клавиши F6 создаю копию файла lab5-1.asm с именем lab5-2.asm. (рис. 7)

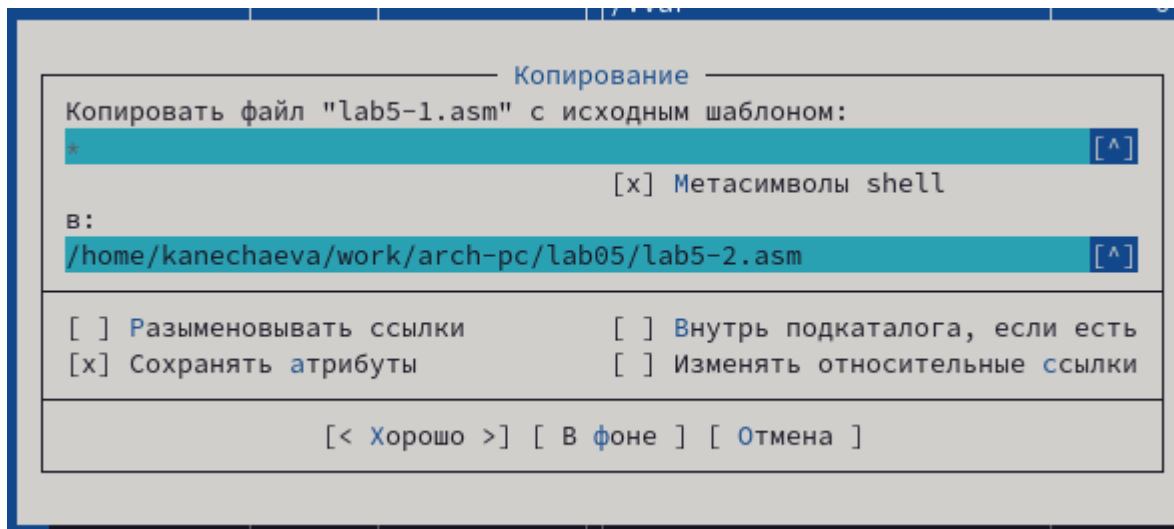


Рис. 7. Копирование файла

Исправляю текст программы в файле lab5-2.asm с использованием подпрограмм из внешнего файла in_out.asm в соответствии с листингом 5.2. (рис. 8)

```
mc [kanechaeva@192.168.1.14]:~/work/arch-pc/lab05
lab5-2.asm [-M--] 41 L:[ 1+13 14/ 14] *(963 / 963b) <EOF>
#include 'in_out.asm' ; подключение внешнего файла
SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку: ',0h ; сообщение
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
mov eax, msg ; запись адреса выводимого сообщения в `EAX`
call sprintf ; вызов подпрограммы печати сообщения
mov ecx, buf1 ; запись адреса переменной в `EAX`
mov edx, 80 ; запись длины вводимого сообщения в `EBX`
call sread ; вызов подпрограммы ввода сообщения
call quit ; вызов подпрограммы завершения
```

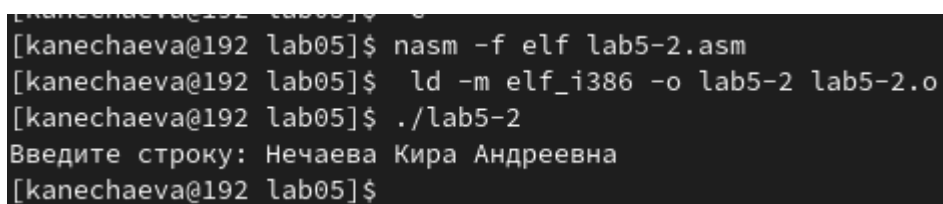
Рис. 8. Изменение файла

Транслирую файл, выполняю компоновку созданного объектного файла, запускаю новый исполняемый файл. (рис. 9)

```
[kanechaeva@192 lab05]$ nasm -f elf lab5-2.asm
[kanechaeva@192 lab05]$ ^C
[kanechaeva@192 lab05]$ ld -m elf_i386 -o lab5-2 lab5-2.o
[kanechaeva@192 lab05]$ ./lab5-2
Введите строку:
Нечаева Кира Андреевна
[kanechaeva@192 lab05]$
```

Рис. 9. Исполнение файла

Теперь в файле lab5-2.asm заменяю подпрограмму sprintLF на sprint. Запускаю его. (рис. 10)



```
[kanechaeva@192 lab05]$ nasm -f elf lab5-2.asm
[kanechaeva@192 lab05]$ ld -m elf_i386 -o lab5-2 lab5-2.o
[kanechaeva@192 lab05]$ ./lab5-2
Введите строку: Нечаева Кира Андреевна
[kanechaeva@192 lab05]$
```

Рис. 10. Запуск файла

Разница в том, что запуск первого файла запрашивает ввод с новой строки, а запуск второго, запрашивает ввод без переноса на новую строку.

4. Задание для самостоятельной работы

1)

Создаю копию файла lab5-1.asm. Вношу изменения в программу листинга 5.1, так чтобы выводила приглашение “Введите строку:”, давала ввести строку с клавиатуры, после чего выводила введённую строку на экран. (рис. 11)

```
lab5-3.asm      [-----]  7  L
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
```

Рис. 11. Редактирование кода

2) Транслирую файл, выполняю компоновку созданного объектного файла, запускаю новый исполняемый файл. (рис. 12)

```
[kanechaeva@192 lab05]$ nasm -f elf lab5-3.asm
[kanechaeva@192 lab05]$ ld -m elf_i386 -o lab5-3 lab5-3.o
[kanechaeva@192 lab05]$ ./lab5-3
Введите строку:
Нечаева Кира Андреевна
Нечаева Кира Андреевна
```

Рис. 12. Запуск файла

3) Создаю копию файла lab5-2.asm. Исправляю текст программы листинг 5.2, так чтобы она выводила приглашение “Введите строку:”, давала ввести строку с клавиатуры, после чего выводила введенную строку на экран. (рис. 13)

```
[lab5-4.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 sprint
mov ecx, buf1
mov edx, 80
call sread
mov eax, 4
mov ebx, 1
mov ecx, buf1
int 80h
call quit
```

Рис. 13. Редактирование файла

4) Транслирую файл, выполняю компоновку созданного объектного файла, запускаю новый исполняемый файл. (рис. 14)

```
[kanechaeva@192 lab05]$ nasm -f elf lab5-4.asm
[kanechaeva@192 lab05]$ ld -m elf_i386 -o lab5-4 lab5-4.o
[kanechaeva@192 lab05]$ ./lab5-4
Введите строку: Нечаева
Нечаева
[kanechaeva@192 lab05]$
```

Рис. 14. Исполнение файла

4 Вывод

При выполнении данной лабораторной работы я приобрела практические навыки работы в Midnight Commander и освоила инструкции языка ассемблера `mov` и `int`.

5 Источники

1. ТУИС – Архитектура ЭВМ – [Электронный ресурс] -
<https://esystem.rudn.ru/mod/resource/view.php?id=1030553>