

# **Шаблон отчёта по лабораторной работе**

5

Разанацуа Сара Естэлл

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>7</b>
<b>4</b>	<b>Выводы</b>	<b>16</b>
	<b>Список литературы</b>	<b>17</b>

## Список иллюстраций

3.1	открытый mc . . . . .	7
3.2	создание каталога . . . . .	8
3.3	перемещение между директориями . . . . .	8
3.4	редактирование файла . . . . .	9
3.5	компиляция файла, передача на обработку компоновку и исполнение файла . . . . .	9
3.6	скачанный и копирование файла . . . . .	10
3.7	копирование файла . . . . .	11
3.8	редактирование файла . . . . .	11
3.9	исполнение файла . . . . .	12
3.10	отредактированный файл . . . . .	12
3.11	исполнение файла . . . . .	12
3.12	копирование файла . . . . .	13
3.13	редактирование файла . . . . .	13
3.14	исполнение файла . . . . .	14
3.15	копирование файла . . . . .	14
3.16	редактирование файла . . . . .	14
3.17	исполнение файла . . . . .	15

## Список таблиц

# 1 Цель работы

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

## 2 Задание

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

### 3 Выполнение лабораторной работы

- 1. Основы работы с mc
- Открываю Midnight Commander, введя в терминал mc. (рис. 3.1).

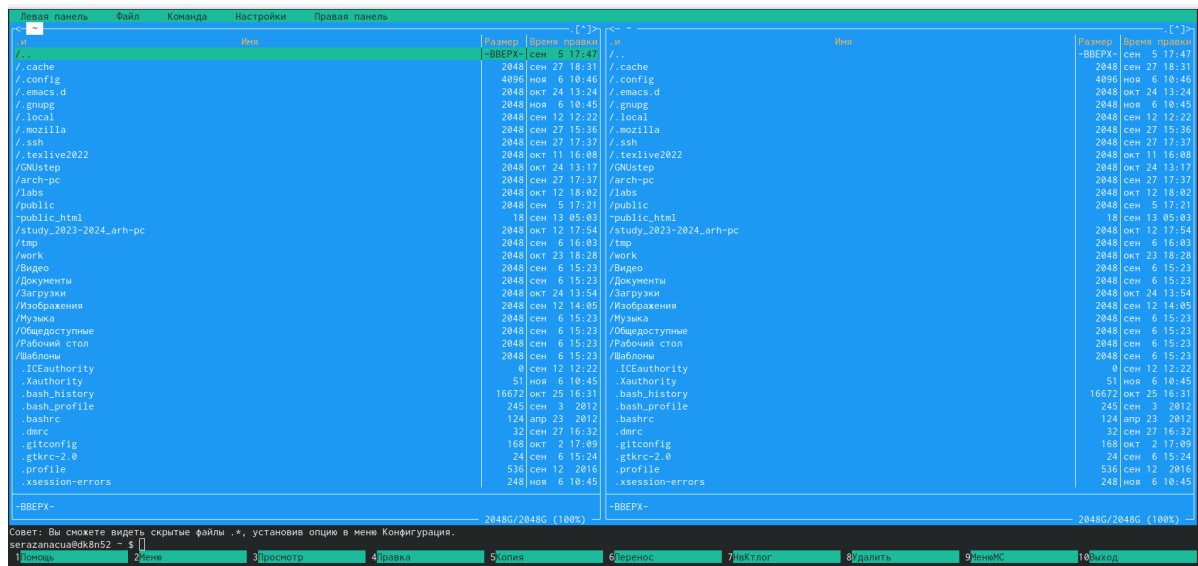


Рис. 3.1: открытый mc

- Перехожу в каталог ~/work/study/2022-2023/Архитектура Компьютера/arch-pc, используя файловый менеджер mc. И с помощью функциональной клавиши F7 создаю каталог lab05.(рис. 3.2).

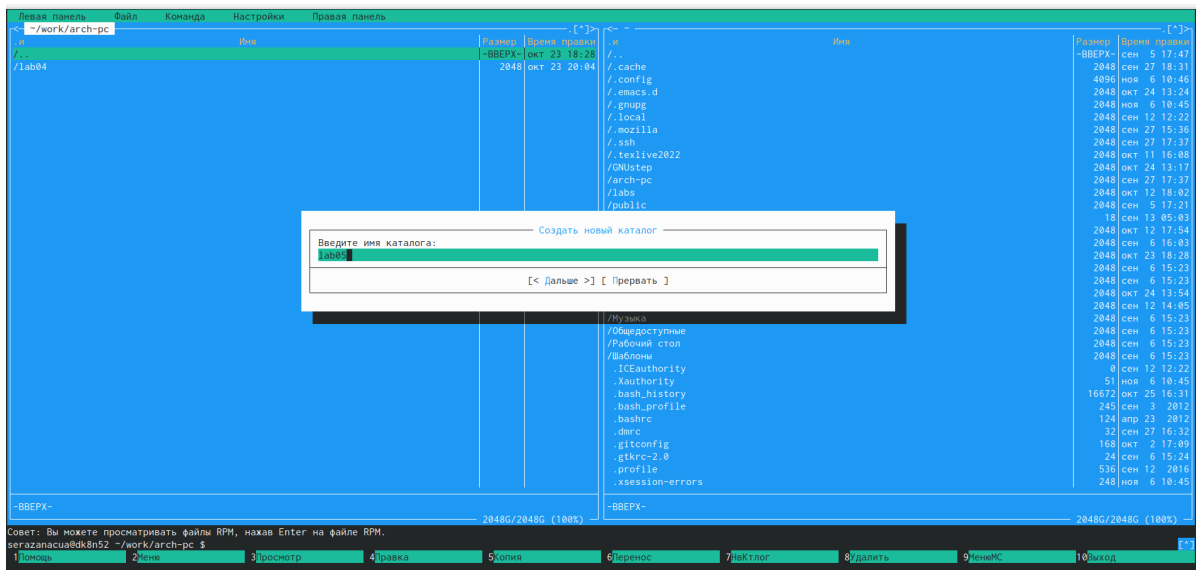


Рис. 3.2: создание каталога

- Переходу в созданный каталог. (рис. 3.3).

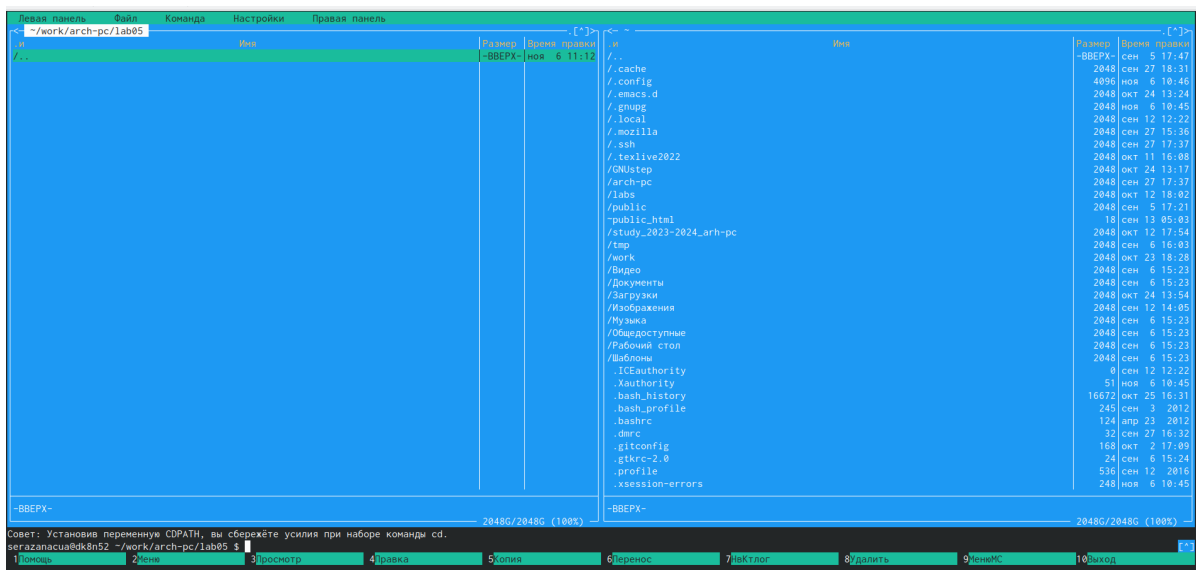
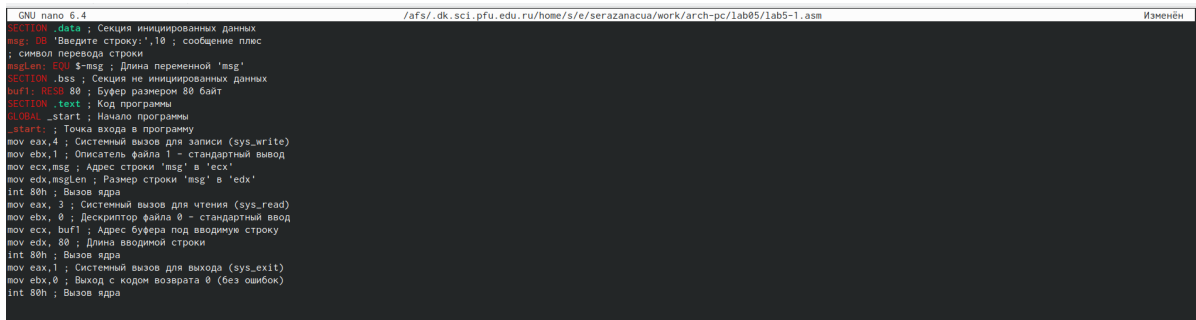


Рис. 3.3: перемещение между директориями

- В строке ввода прописываю команду `touch lab5-1.asm`, чтобы создать файл, в котором буду работать



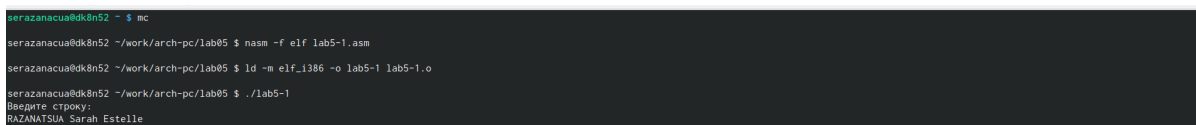
- 2. Структура программы на языке ассемблера NASM
- С помощью функциональной клавиши F4 открываю созданный файл для редактирования в редакторе nano. Ввожу в файл код программы для запроса строки у пользователя (рис. [3.4]). Далее выхожу из файла (Ctrl+X), сохраняя изменения (Y, Enter). (рис. 3.4).



```
GNU nano 6.4 /afs/dk.sci.pfu.edu.ru/home/s/e/serazanacua/work/arch-pc/lab05/lab5-1.asm
SECTION .data ; Секция инициализированных данных
msg DB "Введите строку:",10 ; сообщение плюс
; символ перевода строки
msgLen EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не инициализированных данных
buf: RESB 80 ; Буфер размером 80 байт
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
mov eax,4 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Файловый дескриптор 1 - стандартный вывод
mov ecx,msg ; Адрес строки 'msg' в 'ecx'
mov edx,msgLen ; Размер строки 'msg' в 'edx'
int 80h ; Вызов ядра
mov eax,3 ; Системный вызов для чтения (sys_read)
mov ebx,0 ; Файловый дескриптор 0 - стандартный ввод
mov ecx,buf ; Адрес буфера под вводимую строку
mov edx,80 ; Длина вводимой строки
int 80h ; Вызов ядра
mov eax,1 ; Системный вызов для выхода (sys_exit)
mov ebx,0 ; Выход с кодом возврата 0 (без ошибок)
int 80h ; Вызов ядра
```

Рис. 3.4: редактирование файла

- Транслирую текст программы файла в объектный файл командой `nasm -f elf lab5-1.asm`. Создался объектный файл `lab5-1.o`. Выполняю компоновку объектного файла с помощью команды `ld -m elf_i386 -o lab5-1 lab5-1.o`. Создался исполняемый файл `lab5-1`. И Запускаю исполняемый файл. Программа выводит строку “Введите строку:” и ждет ввода с клавиатуры, я ввожу свои ФИО, на этом программа заканчивает свою работу. (рис. 3.5).



```
serazanacua@dk8n52 ~$ mc
serazanacua@dk8n52 ~/work/arch-pc/lab05 $ nasm -f elf lab5-1.asm
serazanacua@dk8n52 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-1 lab5-1.o
serazanacua@dk8n52 ~/work/arch-pc/lab05 $ ./lab5-1
Введите строку:
RAZANATSLA Sarah Estelle
```

Рис. 3.5: компиляция файла, передача на обработку компоновку и исполнение файла

- 3. Подключение внешнего файла

- Скачиваю файл in\_out.asm со страницы курса в ТУИС. Он сохранился в каталог “Загрузки”. С помощью функциональной клавиши F5 копирую файл in\_out.asm из каталога Загрузки в созданный каталог lab05 (рис. 3.6).

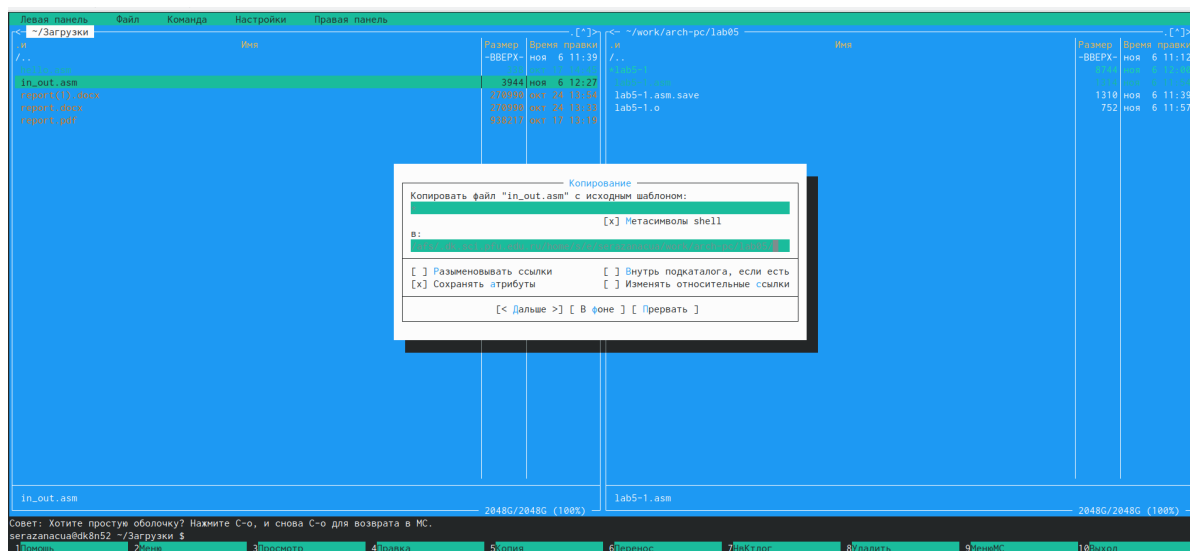


Рис. 3.6: скачанный и копирование файла

- С помощью функциональной клавиши F5 копирую файл lab5-1 в тот же каталог, но с другим именем, для этого в появившемся окне mc прописываю имя для копии файла. (рис. 3.7).

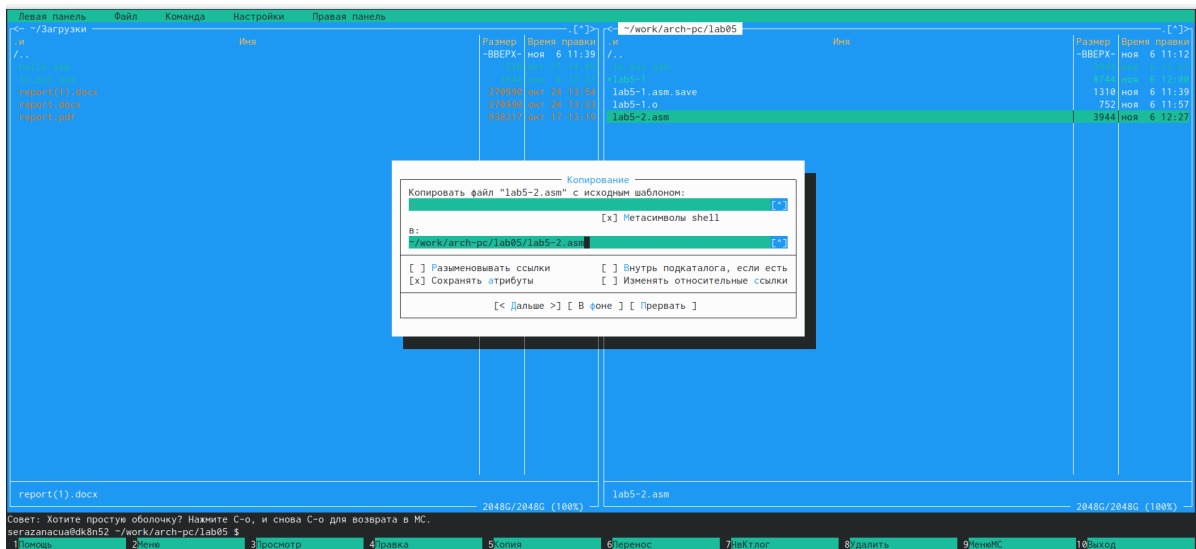


Рис. 3.7: копирование файла

- Изменяю содержимое файла lab5-2.asm во встроенном редакторе nano (рис. [3.8]), чтобы в программе использовались подпрограммы из внешнего файла in\_out.asm. (рис. 3.8).

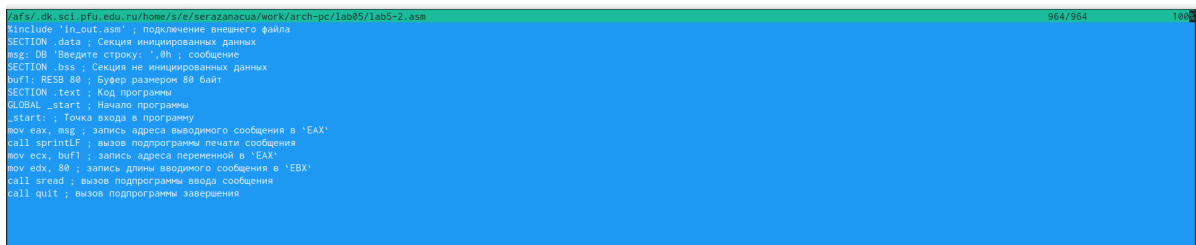


Рис. 3.8: редактирование файла

- Транслирую текст программы файла в объектный файл командой `nasm -f elf lab5-2.asm`. Создался объектный файл lab5-2.o. Выполняю компоновку объектного файла с помощью команды `ld -m elf_i386 -o lab5-2 lab5-2.o`. Создался исполняемый файл lab5-2. Запускаю исполняемый файл. (рис. 3.9).

```

serazanacu@dkn52 ~/work/arch-pc/lab05 $ nasm -f elf lab5-2.asm
serazanacu@dkn52 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-2 lab5-2.o
serazanacu@dkn52 ~/work/arch-pc/lab05 $ ./lab5-2
Введите строку:
RAZANATSUA Sarah Estelle

```

Рис. 3.9: исполнение файла

- Открываю файл lab5-2.asm для редактирования в nano функциональной клавишей F4. Изменяю в нем подпрограмму sprintLF на sprint. Сохраняю изменения и открываю файл для просмотра, чтобы проверить сохранение действий. (рис. 3.10).

```

./lab5-2.o: ELF 32-bit LSB executable, Intel 80386, version 1.0, dynamically linked,
 962/962 100%
#include "input.asm" ; подключение внешнего файла
SECTION .data ; Секция инициализированных данных
msg: DB "Введите строку: ",0h ; сообщение
SECTION .bss ; Секция не инициализированных данных
buf: 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 sprint ; вызов подпрограммы печати сообщения
mov ecx, buf ; запись адреса переменной в 'ECX'
mov edx, 80 ; запись длины вводимого сообщения в 'EDX'
call sread ; вызов подпрограммы ввода сообщения
call quit ; вызов подпрограммы завершения

```

Рис. 3.10: отредактированный файл

- Снова транслирую файл, выполняю компоновку созданного объектного файла, запускаю новый исполняемый файл. (рис. 3.11).

```

serazanacu@dkn52 ~/work/arch-pc/lab05 $ nasm -f elf lab5-2.asm
serazanacu@dkn52 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-2-2 lab5-2.o
serazanacu@dkn52 ~/work/arch-pc/lab05 $ ./lab5-2-2
Введите строку: RAZANATSUA Sarah Estelle
serazanacu@dkn52 ~/work/arch-pc/lab05 $

```

Рис. 3.11: исполнение файла

- Разница между первым исполняемым файлом lab5-2 и вторым lab5-2-2 в том, что запуск первого запрашивает ввод с новой строки, а программа, которая выполняется при запуске второго, запрашивает ввод без переноса на новую строку, потому что в этом заключается различие между подпрограммами sprintLF и sprint.
- 4. Выполнение заданий для самостоятельной работы

- а) Создаю копию файла lab5-1.asm с именем lab5-1-1.asm с помощью функциональной клавиши. (рис. 3.12).

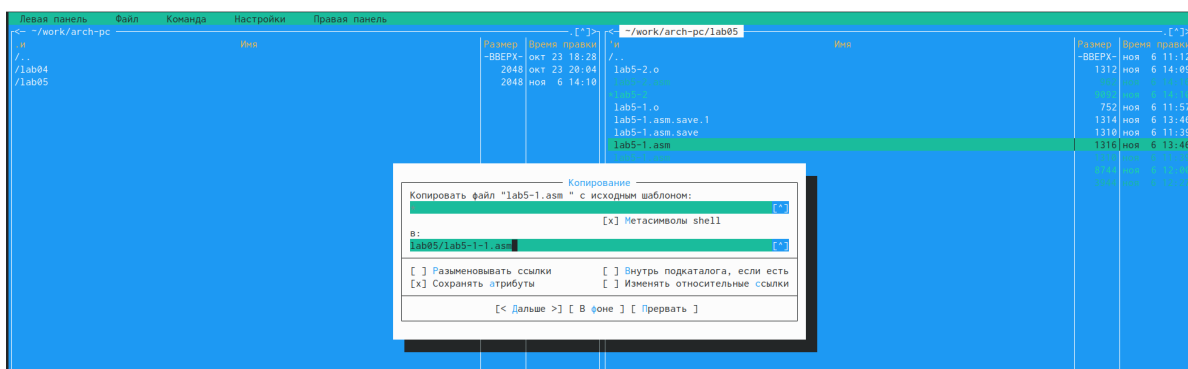


Рис. 3.12: копирование файла

- С помощью функциональной клавиши F4 открываю созданный файл для редактирования. Изменяю программу так, чтобы кроме вывода приглашения и запроса ввода, она выводила вводимую пользователем строку. (рис. 3.13).

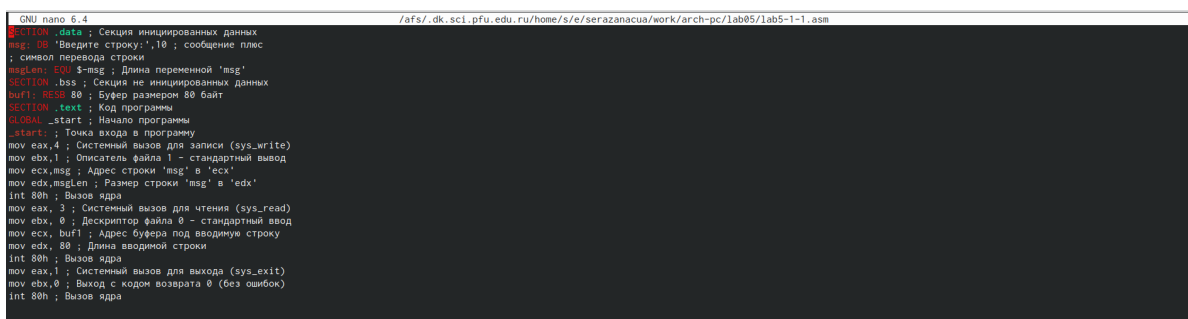


Рис. 3.13: редактирование файла

- б) Создаю объектный файл lab5-1-1.o, отдаю его на обработку компоновщику, получаю исполняемый файл lab5-1-1, запускаю полученный исполняемый файл. Программа запрашивает ввод, ввожу свои ФИО, далее программа выводит введенные мною данные. (рис. 3.14).

```

serazanacu@dkn52 ~/work/arch-pc/lab05 $ nasm -f elf lab5-1-1.asm
serazanacu@dkn52 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-1-1 lab5-1.o
serazanacu@dkn52 ~/work/arch-pc/lab05 $ ./lab5-1-1
Введите строку:
RAZANATSUA Sarah Estelle

```

Рис. 3.14: исполнение файла

- с) Создаю копию файла lab5-2.asm с именем lab5-2-1.asm с помощью функциональной клавиши F5. (рис. 3.15).

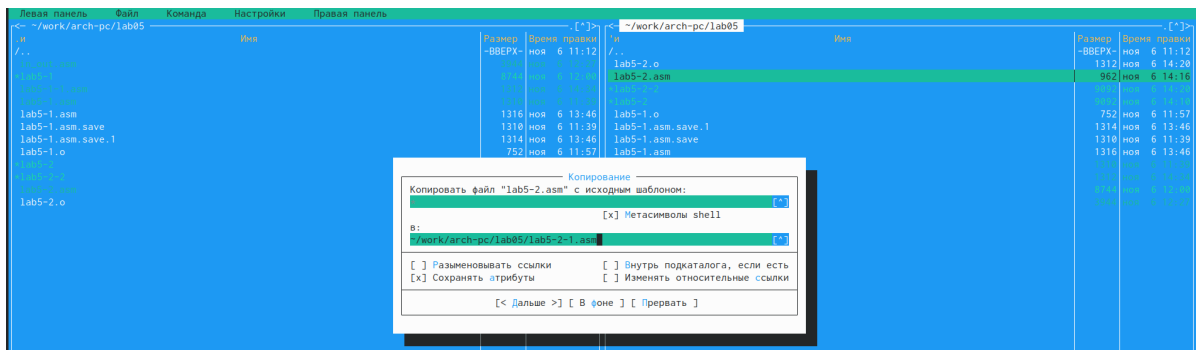


Рис. 3.15: копирование файла

- С помощью функциональной клавиши F4 открываю созданный файл для редактирования. Изменяю программу так, чтобы кроме вывода приглашения и запроса ввода, она выводила вводимую пользователем строку. (рис. 3.16).

```

GNU nano 6.4 /afs/dk.sci.pfu.edu.ru/home/s/e/serazanacu/work/arch-pc/lab05/lab5-2-1.asm
#include "in_out.asm" ; подключение внешнего файла
SECTION .data ; Секция инициализированных данных
msg: db "Введите строку: ", 0Ah ; 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 sprint ; вызов подпрограммы печати сообщения
mov ecx, buf1 ; запись адреса переменной в %ecx
mov edx, 80 ; запись длины вводимого сообщения в %edx
call read ; вызов подпрограммы ввода сообщения
call quit ; вызов подпрограммы завершения

```

Рис. 3.16: редактирование файла

- d) Создаю объектный файл lab5-2-1.o, отдаю его на обработку компоновщику, получаю исполняемый файл lab5-2-1, запускаю полученный исполняемый файл. Программа запрашивает ввод без переноса на новую строку, ввожу свои ФИО, далее программа выводит введенные мною данные. (рис. 3.17).

```
serazanacu@dkn52 ~/work/arch-pc/lab05 $ nasm -f elf lab5-2-1.asm
serazanacu@dkn52 ~/work/arch-pc/lab05 $ ld -m elf_i386 -o lab5-2-1 lab5-2-1.o
serazanacu@dkn52 ~/work/arch-pc/lab05 $ ./lab5-2-1
Введите строку: RAZAMATSUA Sarah Estelle
```

Рис. 3.17: исполнение файла

## 4 Выводы

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



## **Список литературы**