

Отчёта по лабораторной работе №5

Дисциплина: Архитектура компьютера

Камалиева Лия Дамировна.

Содержание

1	Цель работы	5
2	Задание	6
3	Теоретическое введение	7
4	Выполнение лабораторной работы	8
4.1	1.1 Подключение программы	8
4.2	1.2 Подключение внешнего файла in_out.asm	14
4.3	1.3 Задание для самостоятельной работы	16
4.3.1	№1	16
4.3.2	№2	18
4.3.3	№3	19
5	Выводы	23

Список иллюстраций

4.1	мс рис.5.1	8
4.2	Переход в каталог рис.5.2	9
4.3	создание папки рис.5.3	10
4.4	создание asm рис. 5.4	11
4.5	файл рис.5.5	12
4.6	файл2 рис.5.6	13
4.7	компановка рис.5.7	14
4.8	скаченный файл рис.5.8	15
4.9	скаченный файл рис.5.9	16
4.10	создание файла рис.5.10	17
4.11	программа рис.5.11	18
4.12	проверка рис.5.12	19
4.13	создаем файл рис.5.13	20
4.14	замена рис.5.14	21
4.15	проверка рис.5.15	22

Список таблиц

1 Цель работы

Приобретение практических навыков работы в Midnight Commander. Освоение инструкций языка ассемблера `mov` и `int`

2 Задание

- 1.1. Подключение программы
- 1.2. Подключение внешнего файла in_out.asm
- 1.3. Задание для самостоятельной работы

3 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.

4 Выполнение лабораторной работы

4.1 1.1 Подключение программы

Шаг 1. открываем через терминал Midnight Commander

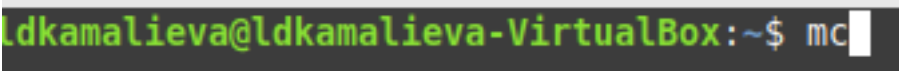
A screenshot of a terminal window with a dark background. The prompt is 'ldkamalieva@ldkamalieva-VirtualBox: ~\$' in green text. The command 'mc' is entered in white text, followed by a white cursor block.

Рис. 4.1: mc рис.5.1

Шаг 2. Переходим в каталог ~/work/arch-rc

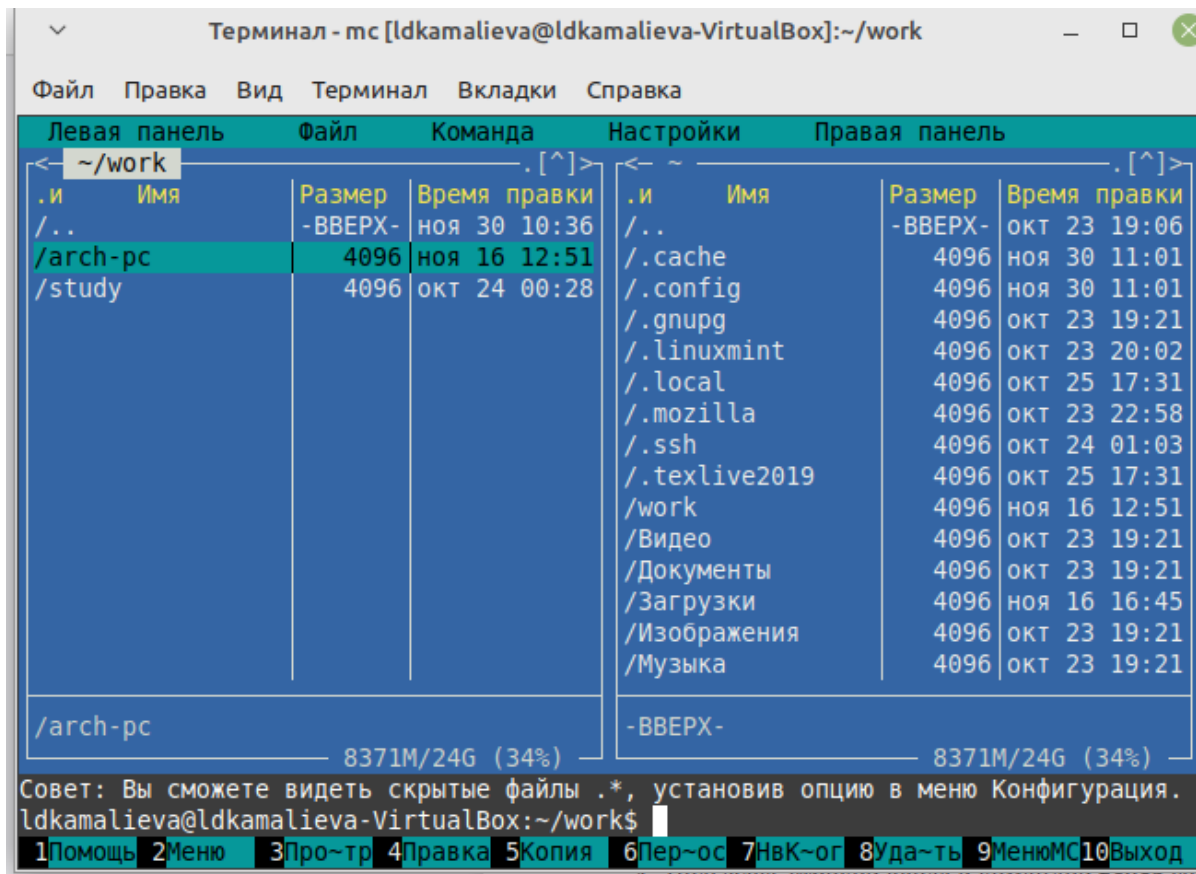


Рис. 4.2: Переход в каталог рис.5.2

Шаг 3. С пощи клавиши (у меня на ноутбуке) Fn + F7, создаем папку lab05

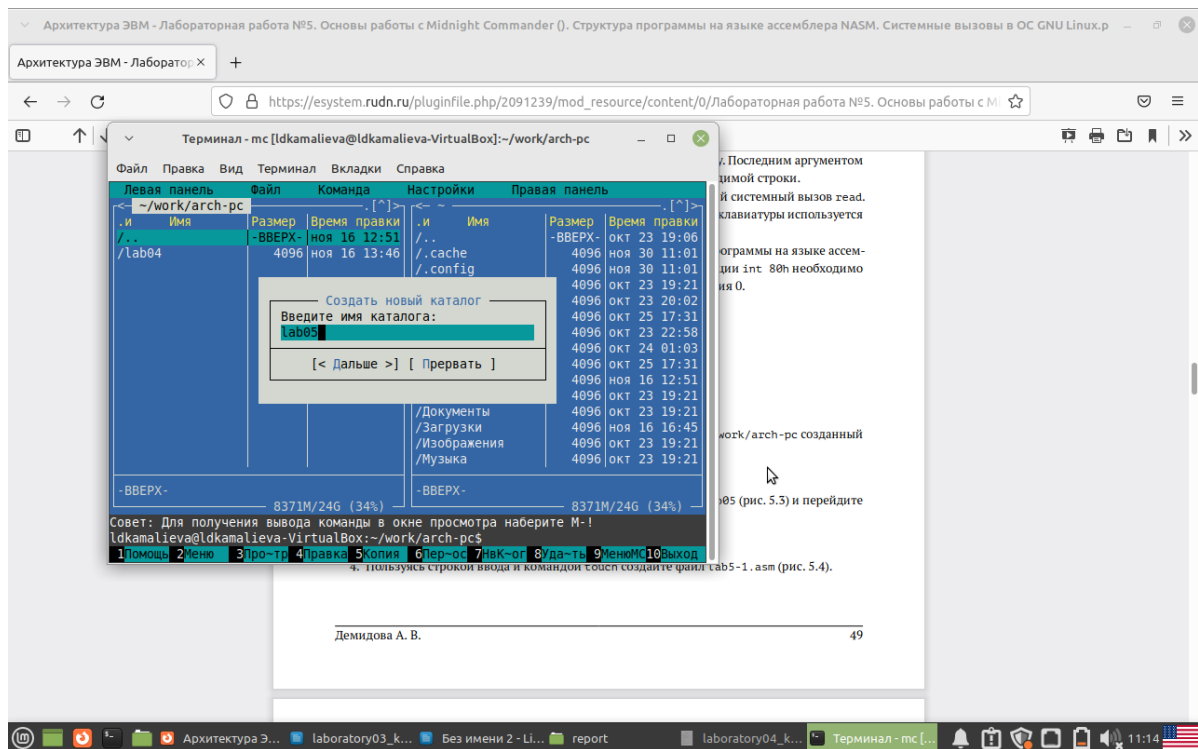


Рис. 4.3: создание папки рис.5.3

Шаг 4. Используем команду touch для создания lab5-1.asm

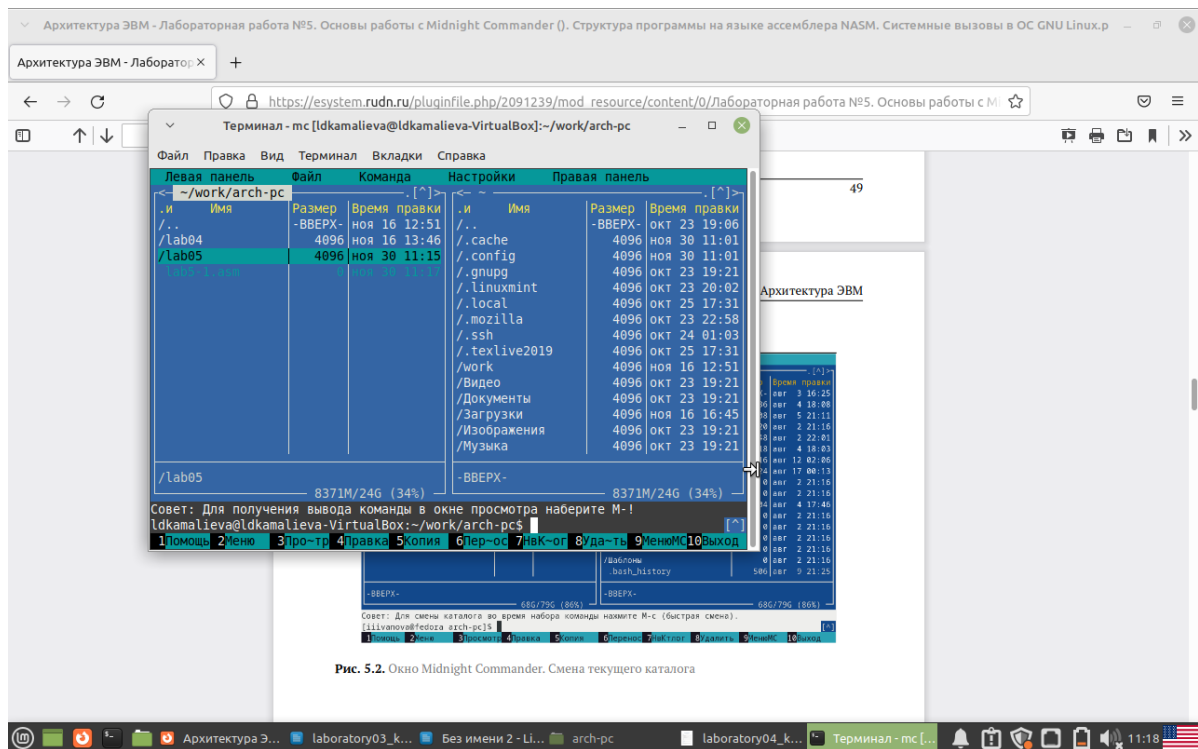


Рис. 4.4: создание asm рис. 5.4

Шаг 5. открываем файл lab5-1.asm и введем текст программы и убедимся, что файл содержит текст программы

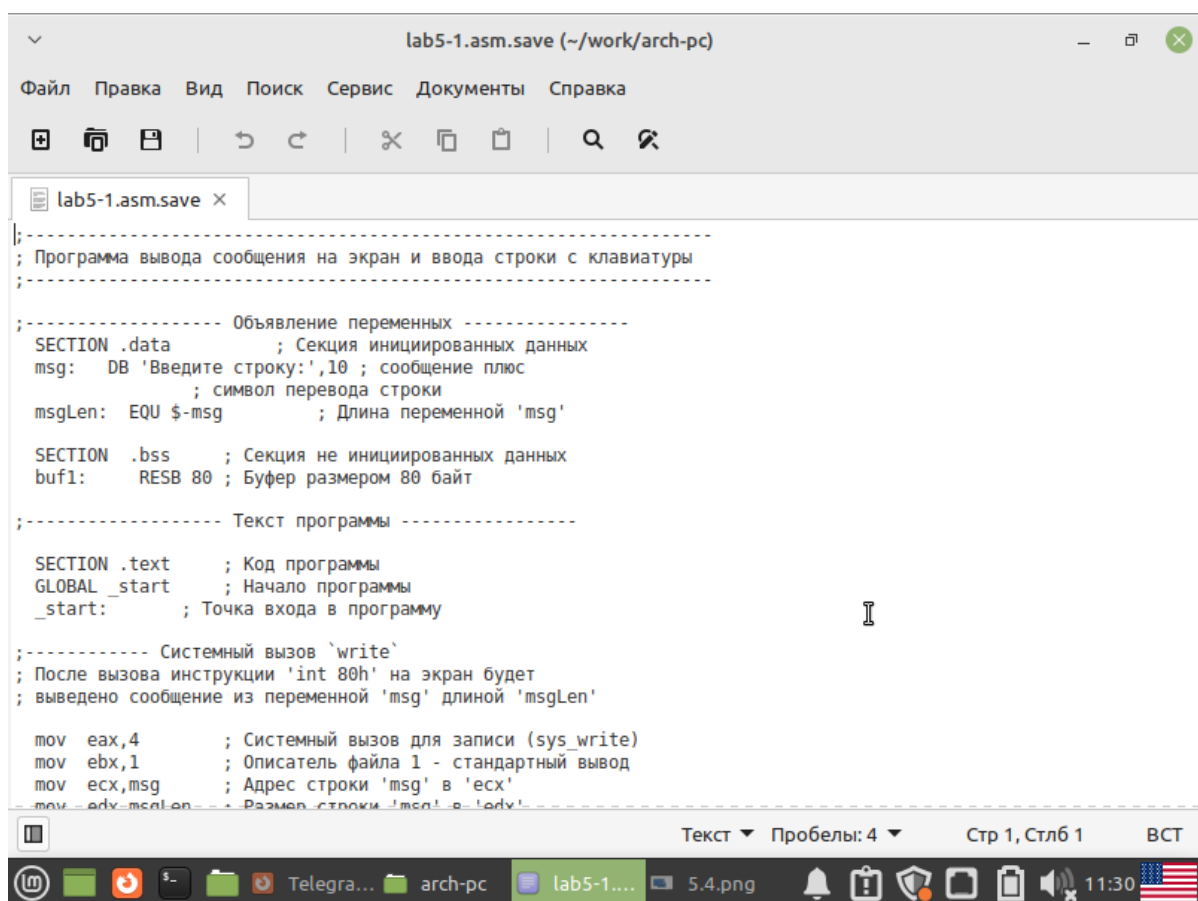


Рис. 4.5: файл рис.5.5

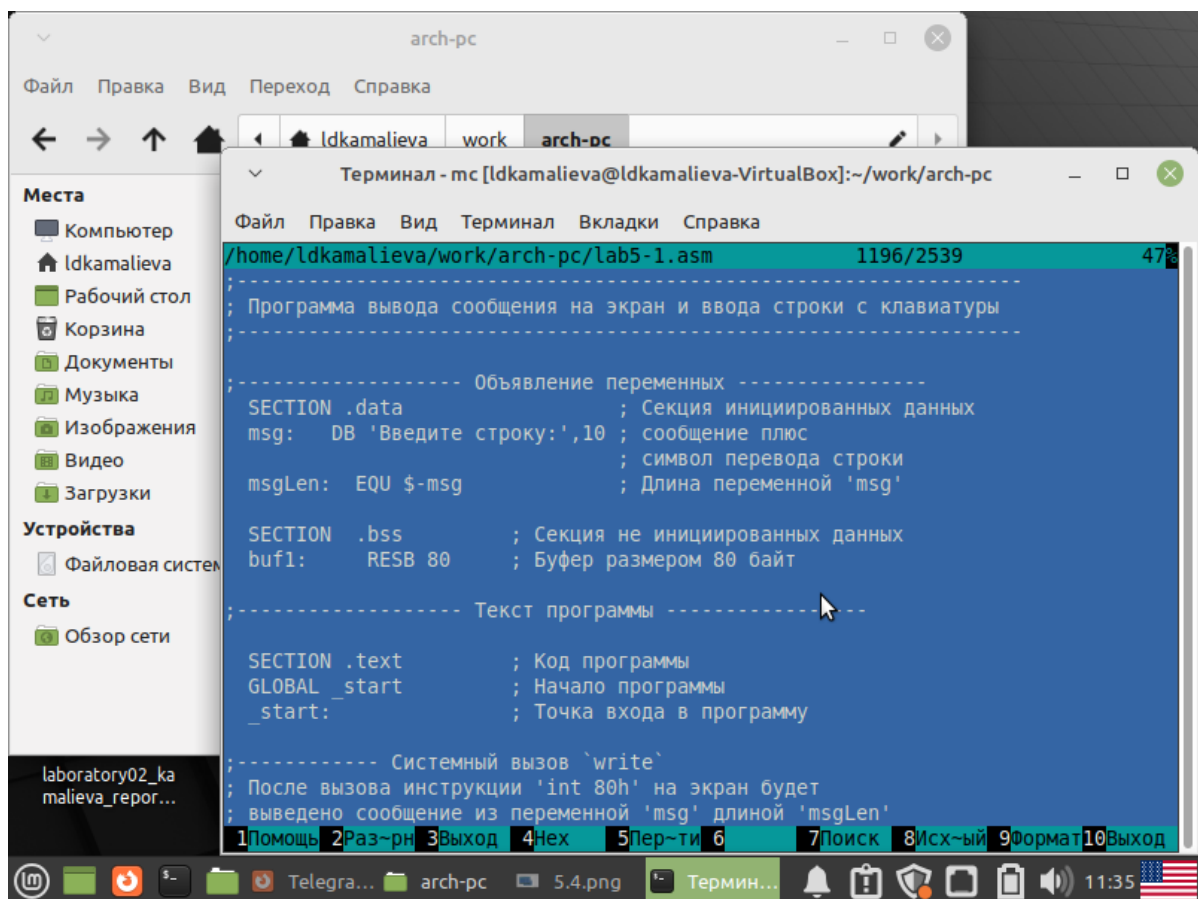


Рис. 4.6: файл2 рис.5.6

Шаг 6. Оттранслируем текст программы lab5-1.asm в объектный файл. Выполним компоновку объектного файла и запустим получившийся исполняемый файл. Программа выводит строку 'Введите строку:' и ожидает ввода с клавиатуры. На запрос вводим своё ФИО

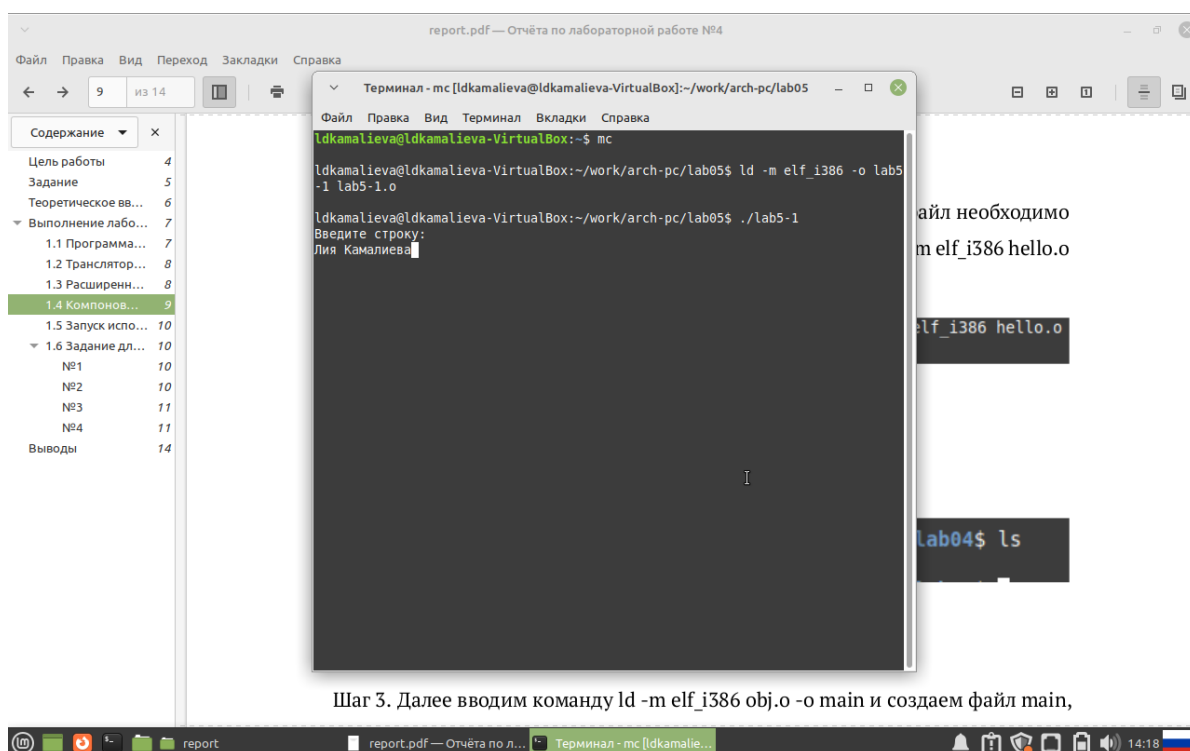


Рис. 4.7: компоновка рис.5.7

4.2 1.2 Подключение внешнего файла in_out.asm

Шаг 1. Скачиваем файл in_out.asm со страницы курса в ТУИС. Подключаемый файл должен лежать в том же каталоге, что и файл с программой, в которой он используется. С помощью функциональной клавиши F6 создаём копию файла lab5-1.asm с именем lab5-2.asm. Исправляем текст программы в файле lab5-2.asm с использованием подпрограмм из внешнего файла in_out.asm. В файле lab5-2.asm заменяем подпрограмму sprintLF на sprint.

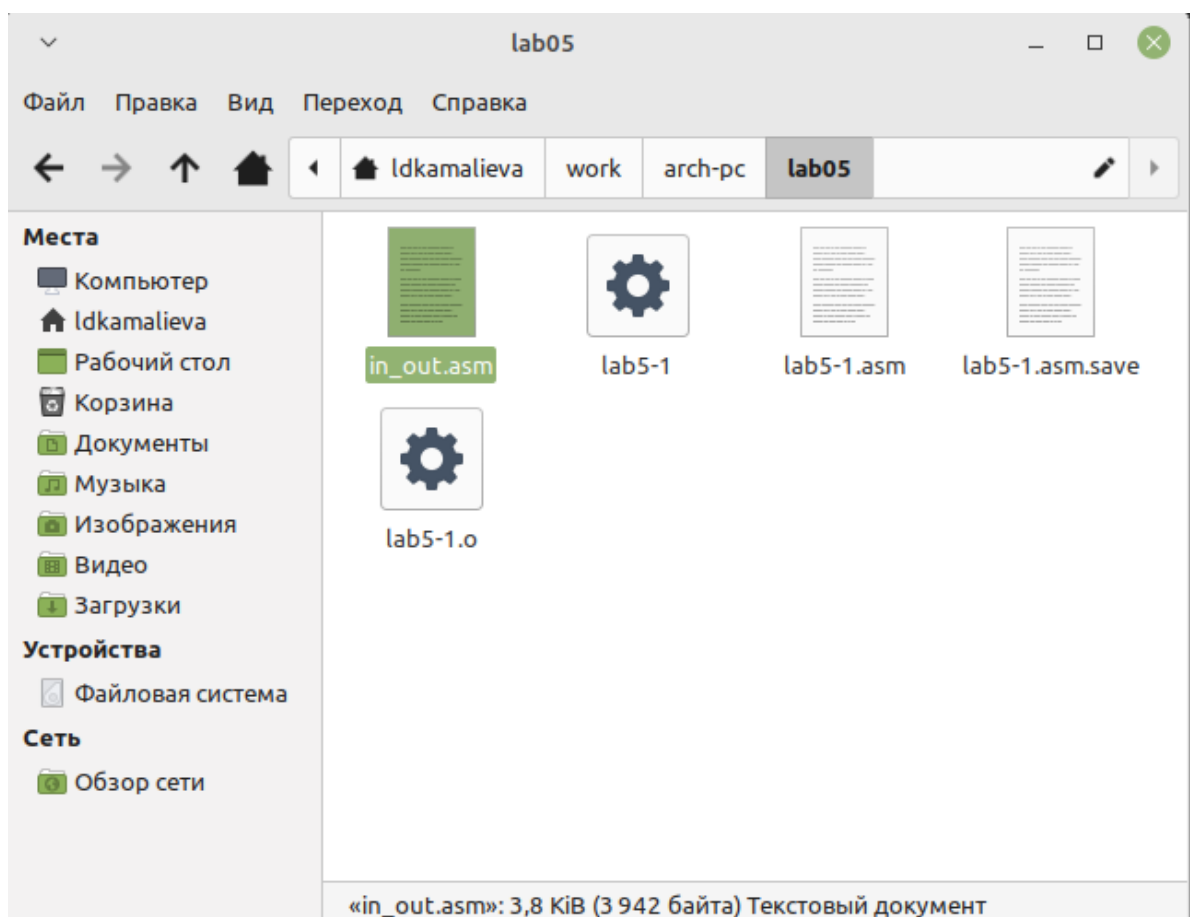
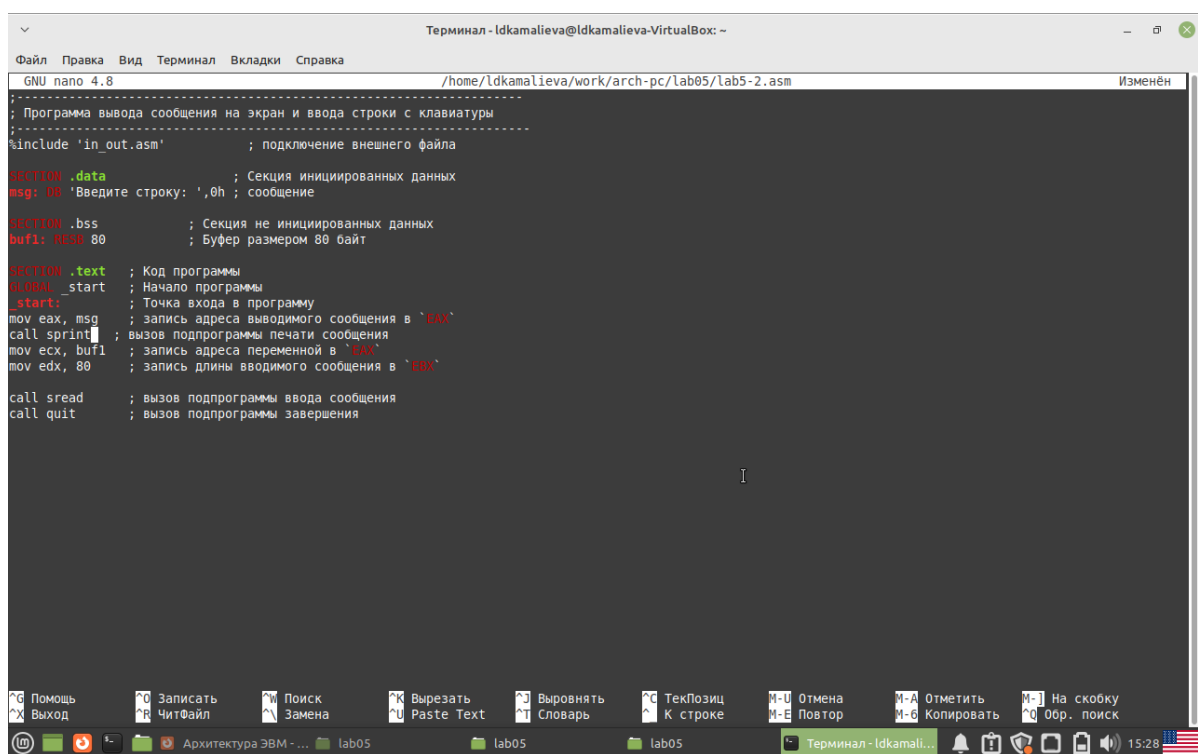


Рис. 4.8: скаченный файл рис.5.8



The screenshot shows a terminal window titled "Терминал - ldkamalieva@ldkamalieva-VirtualBox: ~". The window contains the GNU nano 4.8 editor editing the file `/home/ldkamalieva/work/arch-pc/lab05/lab5-2.asm`. The code is an assembly program for x86-64, using the `syscalls` system call interface. It includes a message `msg: DB 'Введите строку: ', 0h` and a buffer `buf1: RDB 80`. The program uses `sprint` to print the message and `sread` to read input from the user. The code is as follows:

```
-----  
; Программа вывода сообщения на экран и ввода строки с клавиатуры  
-----  
%include 'in_out.asm' ; подключение внешнего файла  
  
SECTION .data ; Секция иницированных данных  
msg: DB 'Введите строку: ', 0h ; сообщение  
  
SECTION .bss ; Секция не иницированных данных  
buf1: RDB 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 sread ; вызов подпрограммы ввода сообщения  
call quit ; вызов подпрограммы завершения
```

The terminal window also shows a menu bar with options like "Помощь", "Записать", "Поиск", "Вырезать", "Вывод", "Словарь", "ТекПозиц", "Отмена", "Отметить", "На скобку", and "Обр. поиск". The status bar at the bottom shows the current directory as `lab05` and the time as 15:28.

Рис. 4.9: скаченный файл рис.5.9

4.3 1.3 Задание для самостоятельной работы

4.3.1 №1

Шаг 1. Создаем копию файла `lab5-1.asm`. Вносим изменения в программу.

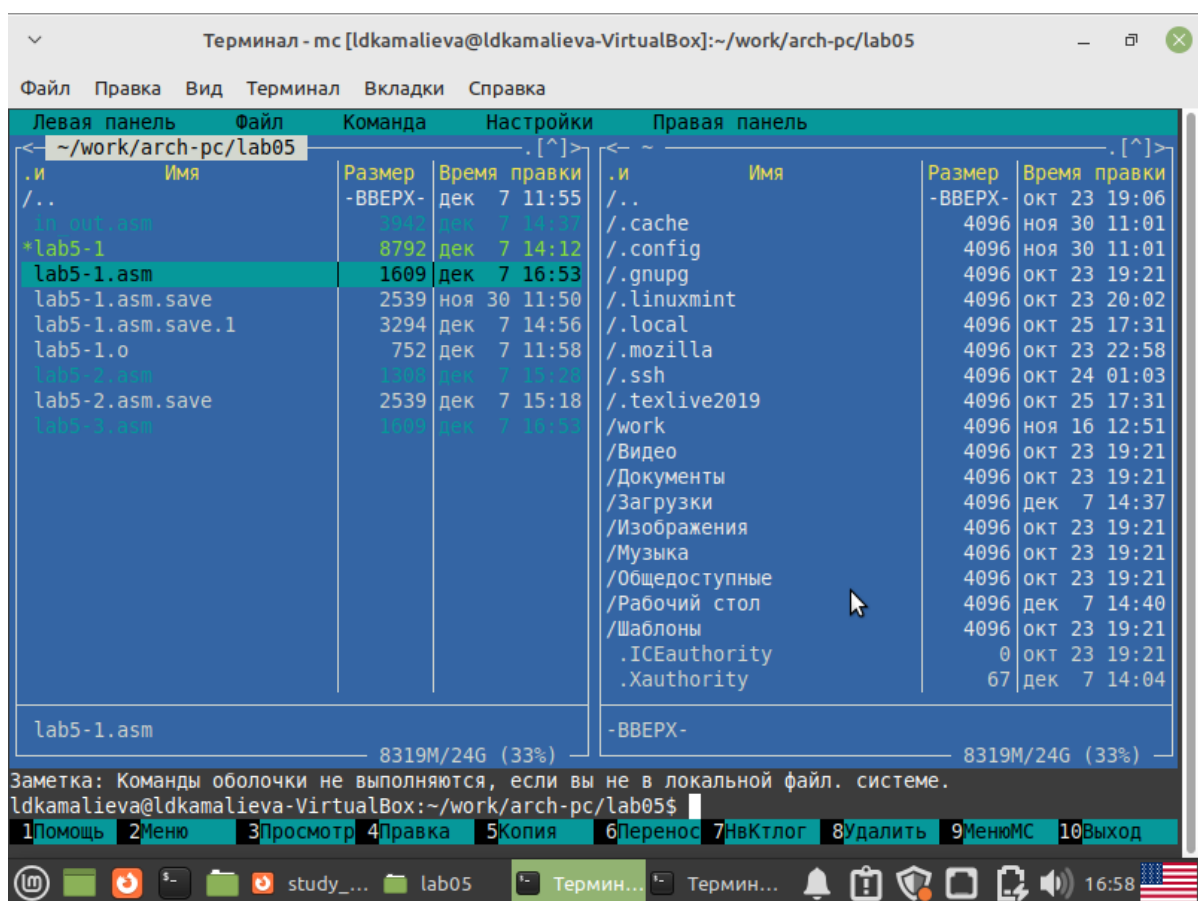


Рис. 4.10: создание файла рис.5.10

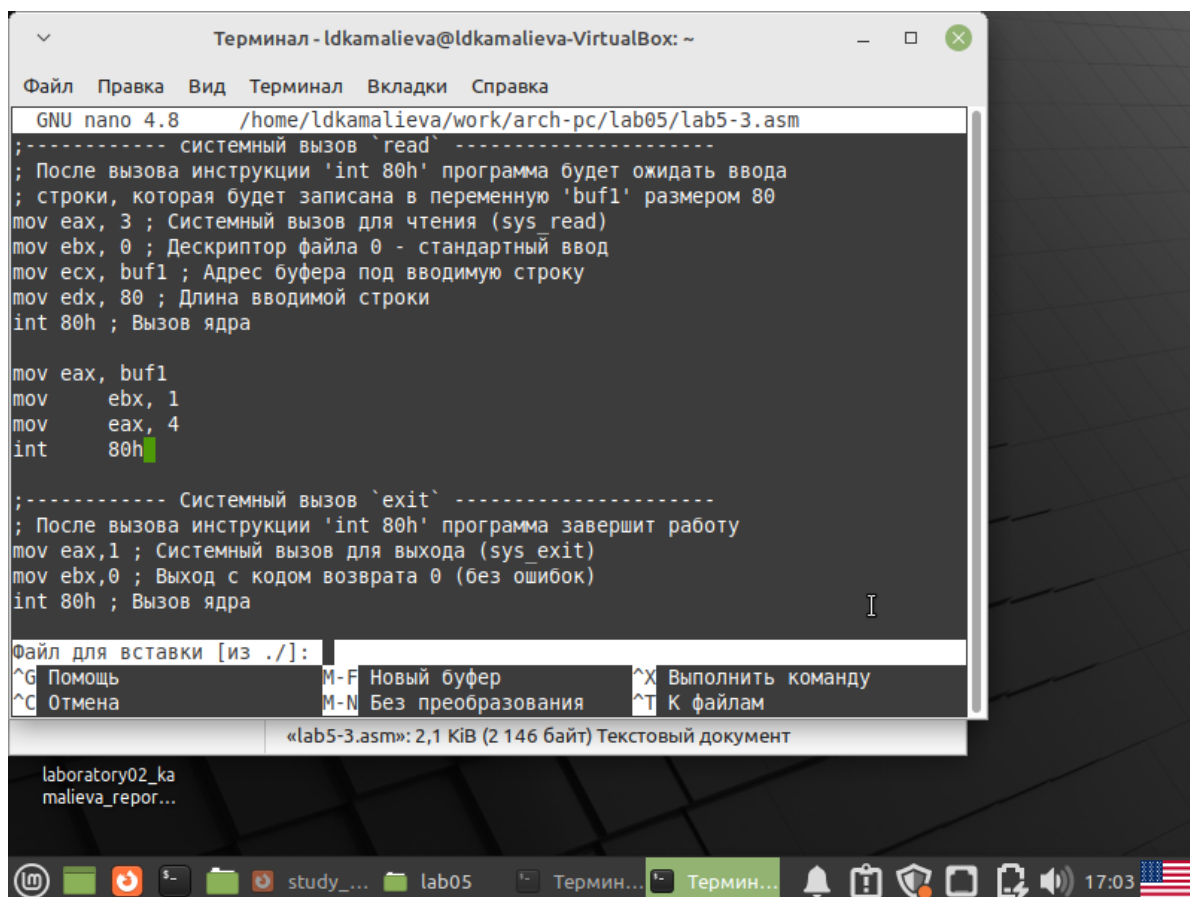


Рис. 4.11: программа рис.5.11

4.3.2 №2

Шаг 1. Делаем проверку. На приглашение ввести строку вводим свое имя и фамилию

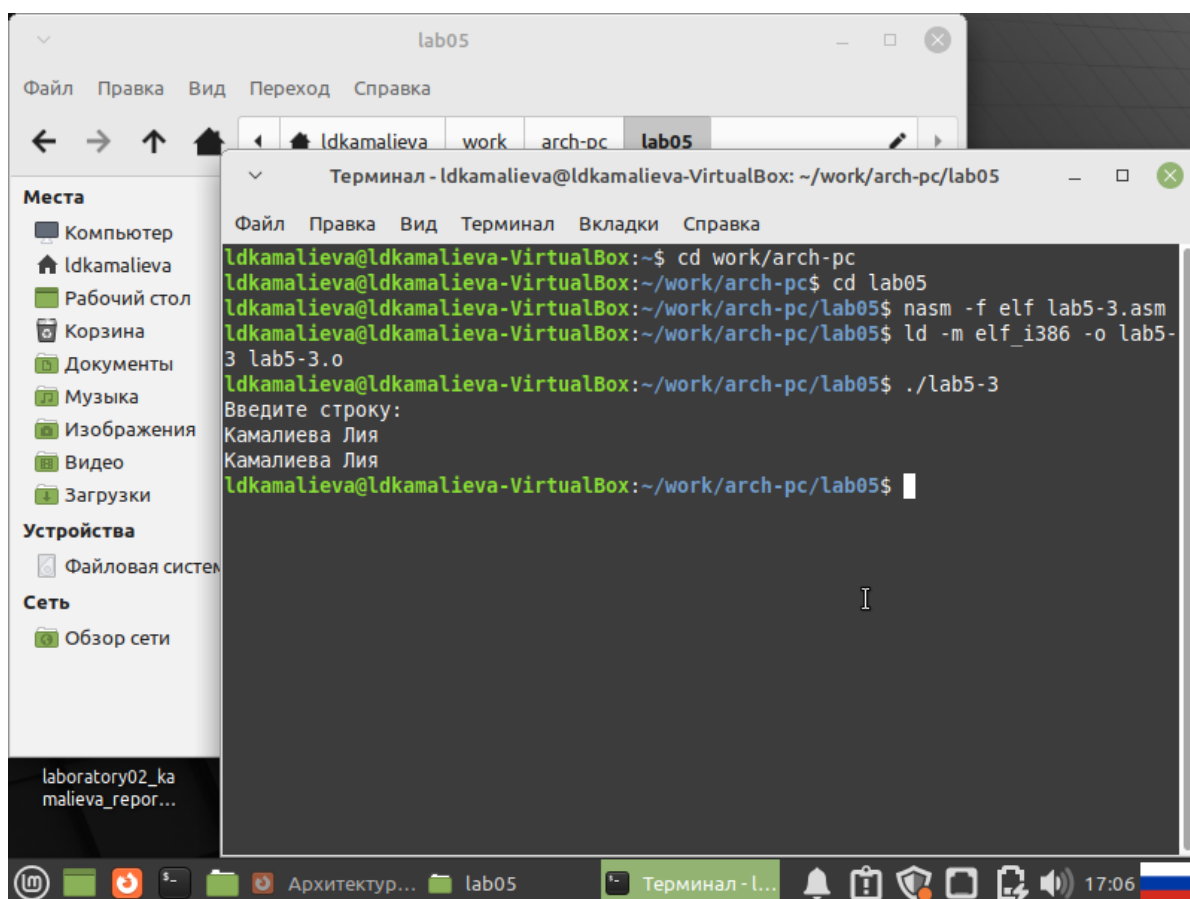


Рис. 4.12: проверка рис.5.12

4.3.3 №3

Создаем копию файла lab5-2.asm. И исправляем текст программы с использование подпрограмм из внешнего файла in_out.asm, так чтобы она работала по следующему алгоритму: • вывести приглашение типа “Введите строку:”; • ввести строку с клавиатуры; • вывести введенную строку на экран.

Шаг 1. Создаем файл lab5-4

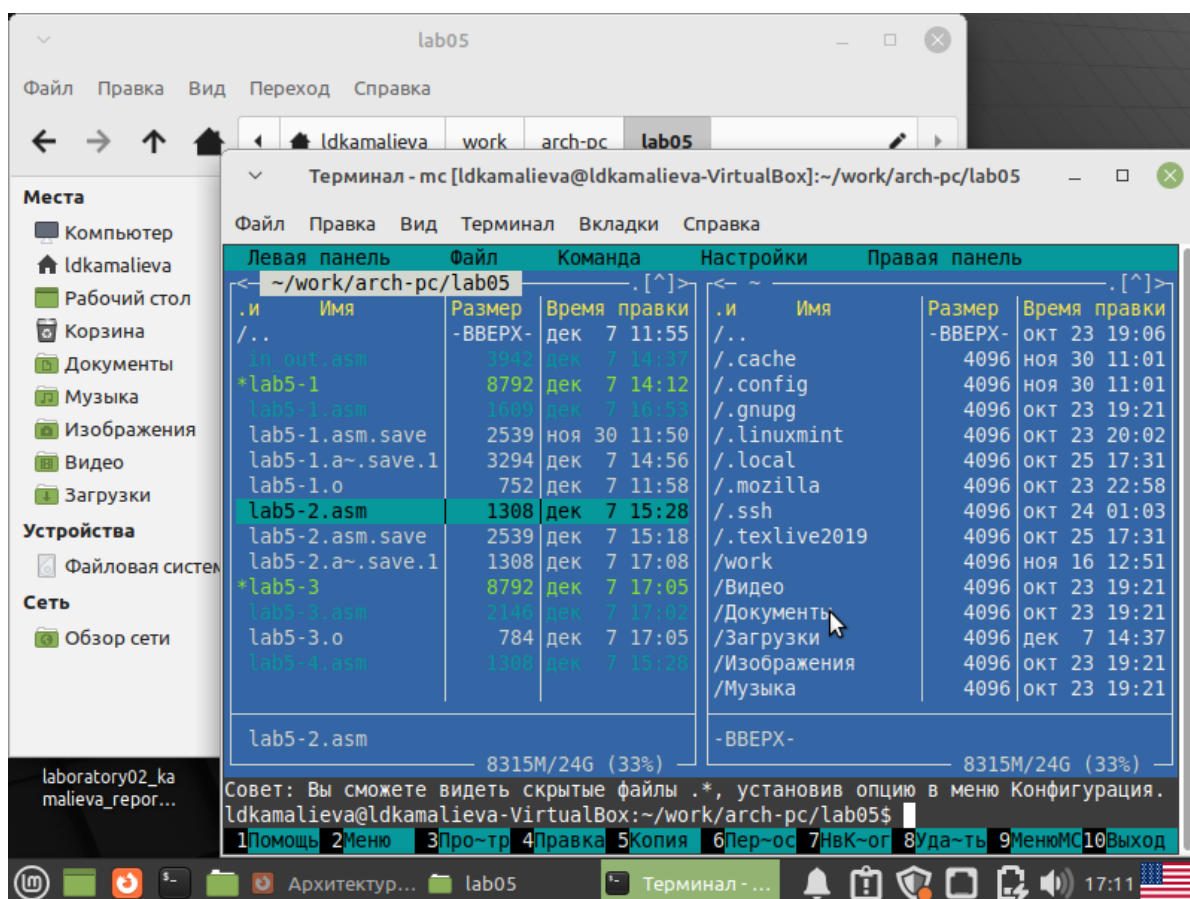
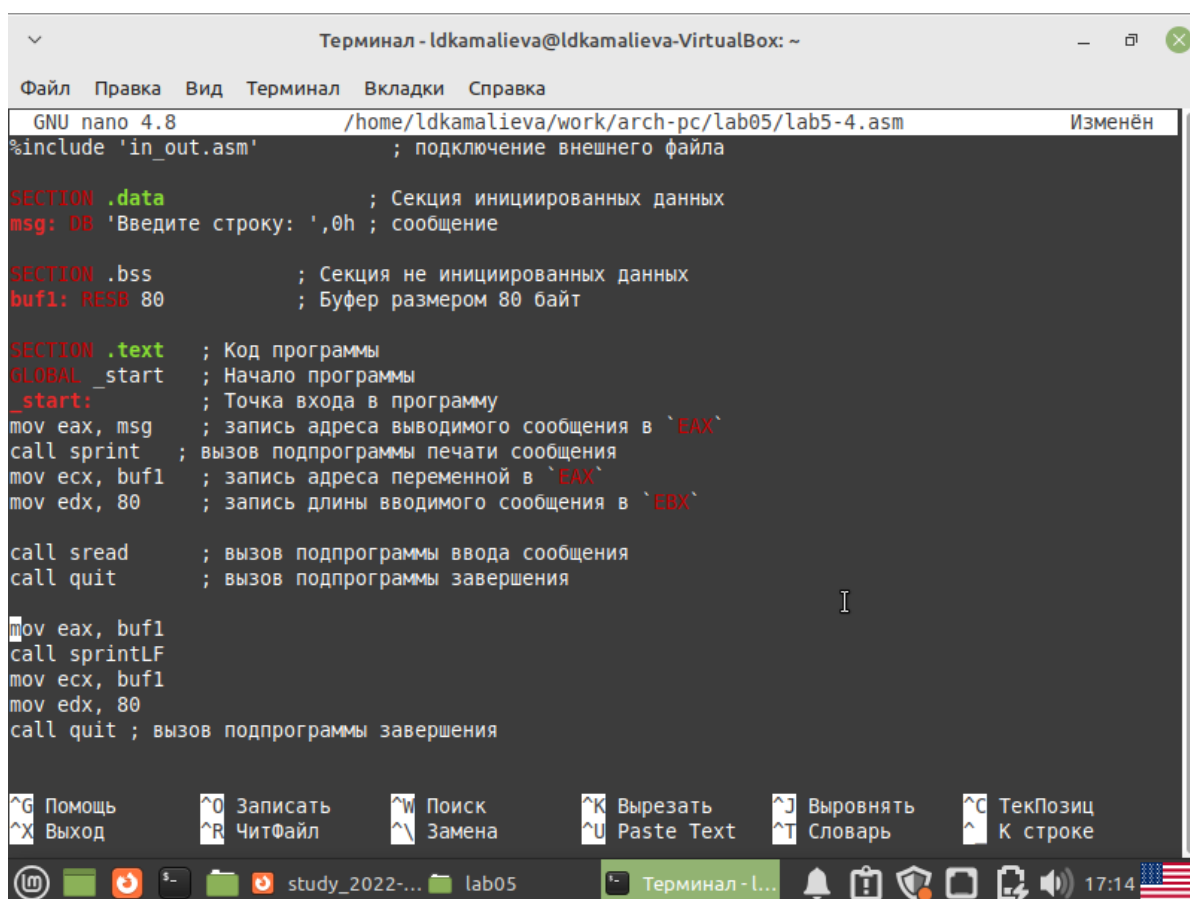


Рис. 4.13: создаем файл рис.5.13

Шаг 2. Изменяем код



```
GNU nano 4.8 /home/ldkamaliev/work/arch-pc/lab05/lab5-4.asm
%include 'in_out.asm' ; подключение внешнего файла

SECTION .data ; Секция иницированных данных
msg: DB 'Введите строку: ',0h ; сообщение

SECTION .bss ; Секция не иницированных данных
buf1: RESB 80 ; Буфер размером 80 байт

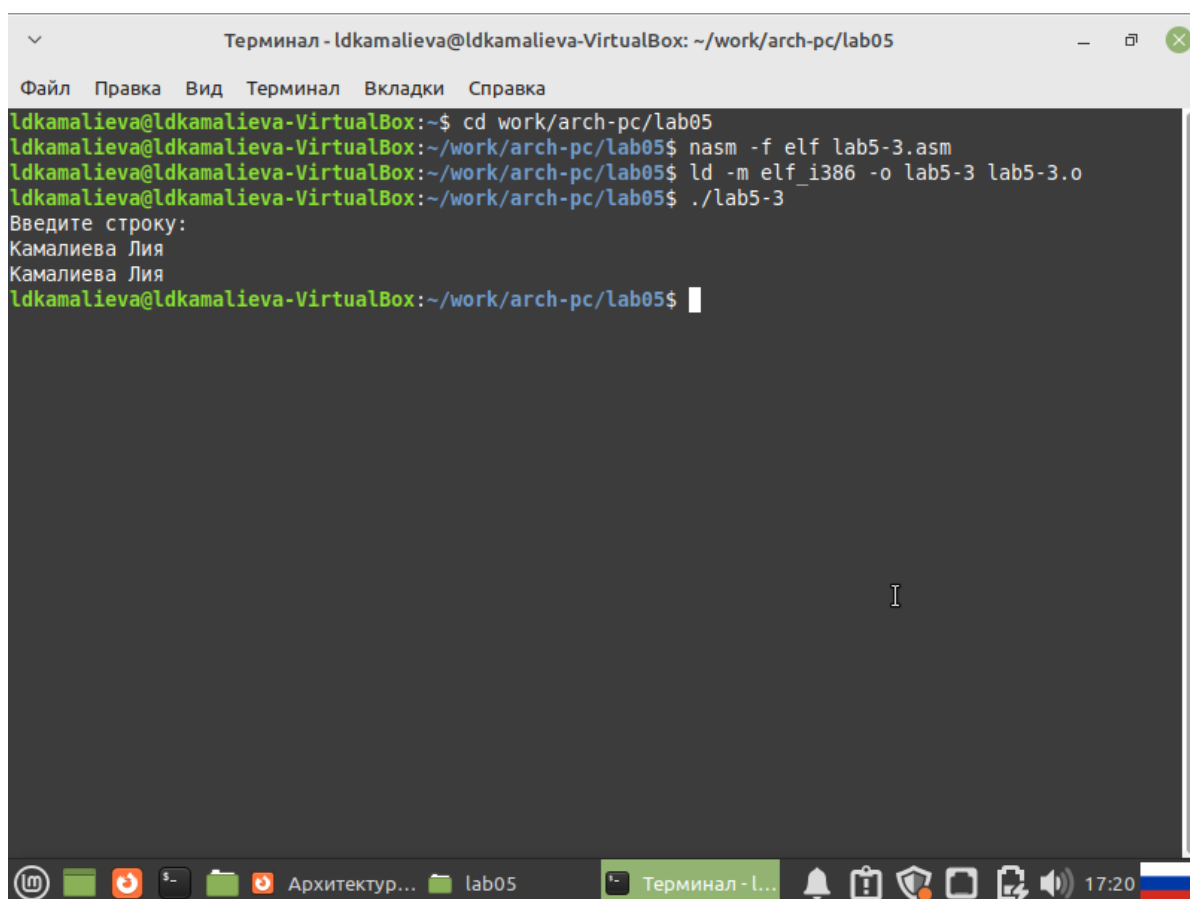
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
mov eax, msg ; запись адреса выводимого сообщения в `EAX`
call sprint ; вызов подпрограммы печати сообщения
mov ecx, buf1 ; запись адреса переменной в `EAX`
mov edx, 80 ; запись длины вводимого сообщения в `EBX`

call sread ; вызов подпрограммы ввода сообщения
call quit ; вызов подпрограммы завершения

mov eax, buf1
call sprintLF
mov ecx, buf1
mov edx, 80
call quit ; вызов подпрограммы завершения
```

Рис. 4.14: замена рис.5.14

Шаг 3. Делаем проверку



```
Терминал - ldkamalieva@ldkamalieva-VirtualBox: ~/work/arch-pc/lab05
Файл  Правка  Вид  Терминал  Вкладки  Справка
ldkamalieva@ldkamalieva-VirtualBox:~$ cd work/arch-pc/lab05
ldkamalieva@ldkamalieva-VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab5-3.asm
ldkamalieva@ldkamalieva-VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-3 lab5-3.o
ldkamalieva@ldkamalieva-VirtualBox:~/work/arch-pc/lab05$ ./lab5-3
Введите строку:
Камалиева Лия
Камалиева Лия
ldkamalieva@ldkamalieva-VirtualBox:~/work/arch-pc/lab05$
```

Рис. 4.15: проверка рис.5.15

5 Выводы

В ходе выполнения лабораторной работы мы научились работать в Midnight Commander.