

РОССИЙСКИЙ УНИВЕРСИТЕТ ДРУЖБЫ НАРОДОВ

Факультет физико-математических и естественных наук

Кафедра прикладной информатики и теории вероятностей

ОТЧЕТ

ПО ЛАБОРАТОРНОЙ РАБОТЕ №4

дисциплина: Архитектура компьютера

Студент: Резвых Никита

Группа: НКАбд-04-24

МОСКВА

2024

## **Содержание**

### **1 Цель работы**

### **2 Задание**

### **3 Теоретическое введение**

### **4 Выполнение лабораторной работы**

#### **4.1 Основы работы с Midnight Commander**

#### **4.2 Работа в NASM**

#### **4.3 Подключение внешнего файла**

#### **4.4 Задание для самостоятельной работы**

### **5 Выводы**

### **Список литературы**

# **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 mc
2. Структура программы на языке ассемблера NASM
3. Подключение внешнего файла
4. Выполнение заданий 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. 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 (учетверённое 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 операнд dst — приёмник, а src — источник. В качестве операнда 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).

## 4 Выполнение лабораторной работы

### 4.1 Основы работы с Midnight Commander

Введя соответствующую команду в терминале (рис. 4.1), я открываю Midnight Commander (рис. 4.2).

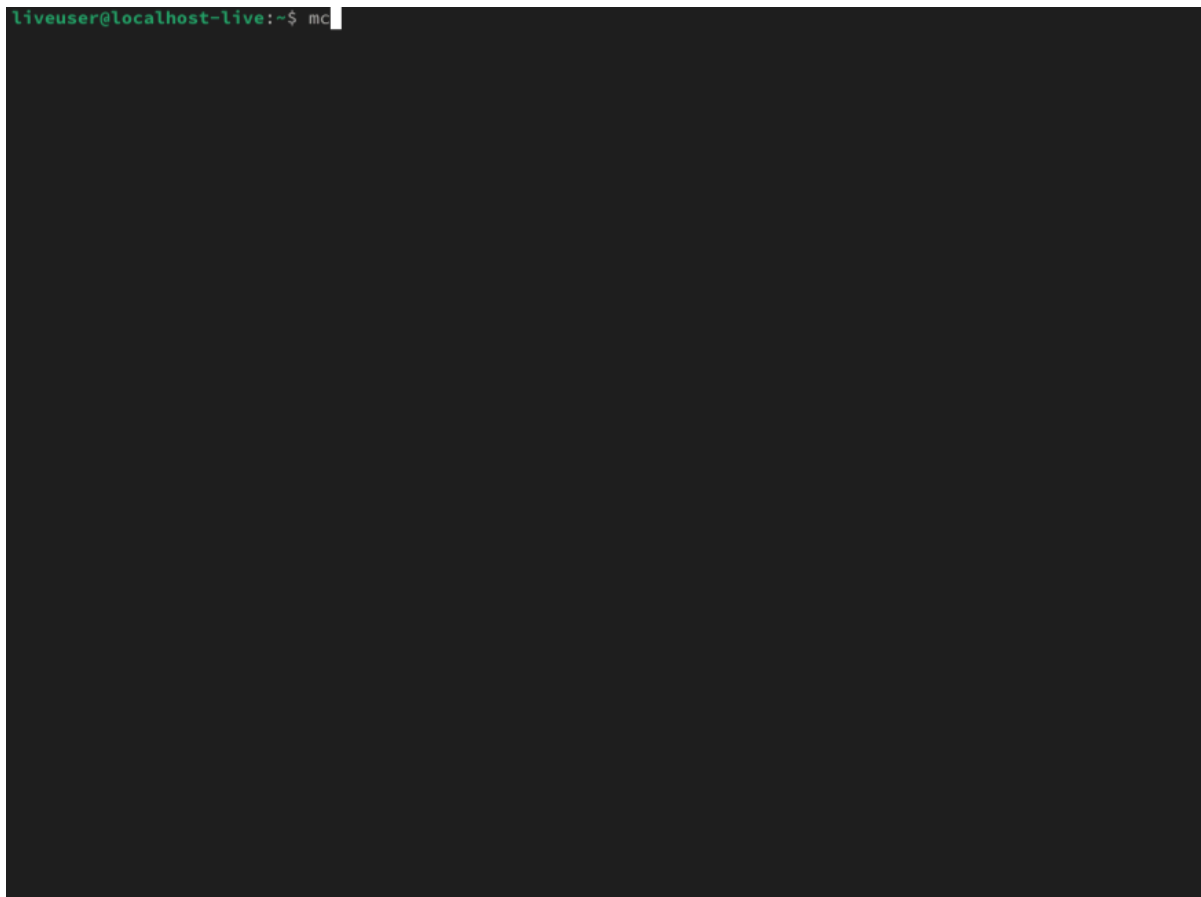


Рис. 4.1: Открытие Midnight Commander



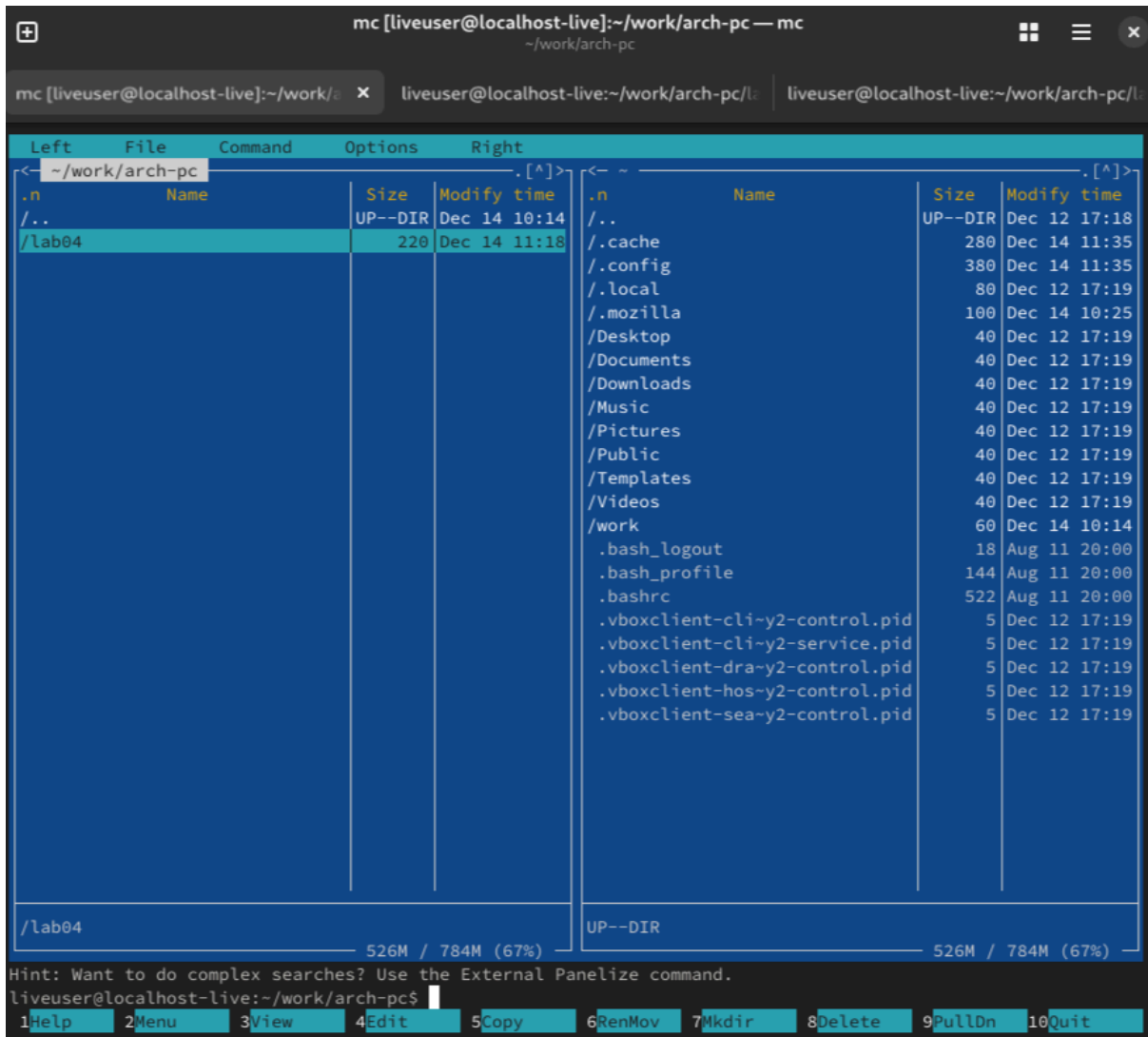


Рис. 4.3: Открытый каталог arch-pc

С помощью функциональной клавиши, я создаю подкаталог lab05, в котором буду работать (рис. 4.4).



В строке ввода ввожу команду touch и создаю файл (рис. 4.5).

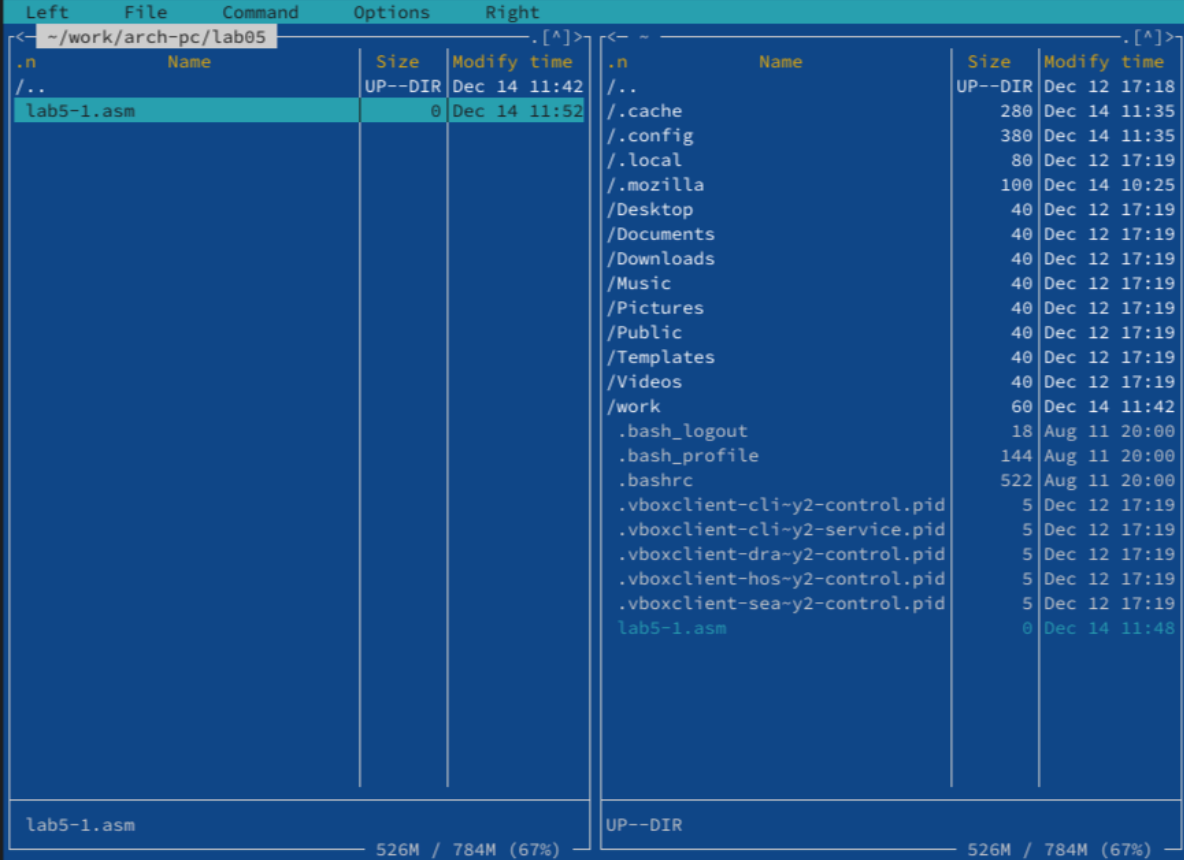
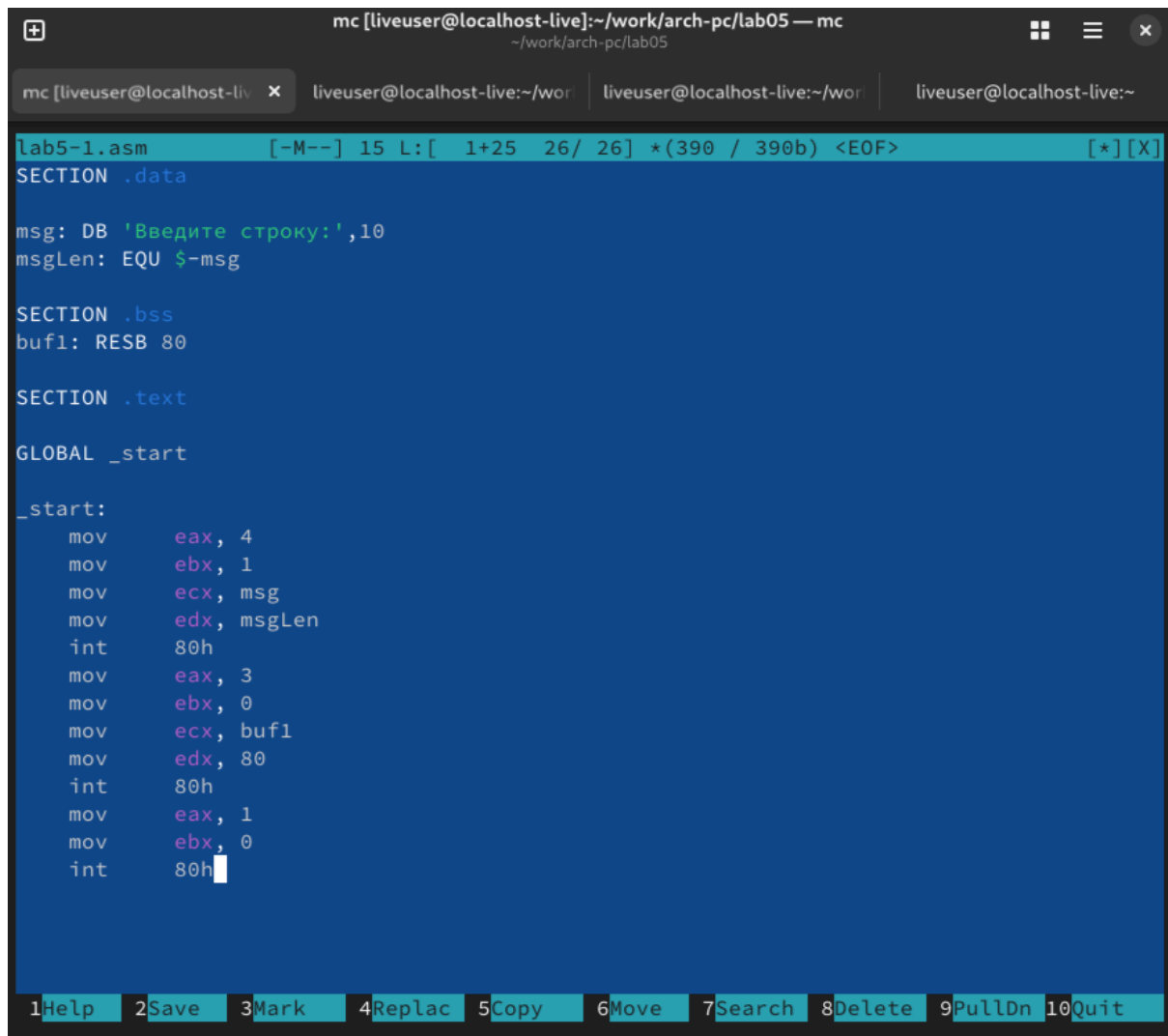


Рис. 4.5: Создание файла в Midnight Commander

## 4.2 Работа в NASM

С помощью F4 открываю только что созданный файл и вношу код с листинга (рис. 4.6).



The screenshot shows the Midnight Commander (mc) interface. The title bar indicates the current directory is `~/work/arch-pc/lab05`. The top status bar shows the file `lab5-1.asm` is open, with a line number of 15 and a cursor at column 26. The main editing area contains the following assembly code:

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

SECTION .bss
buf1: RESB 80

SECTION .text

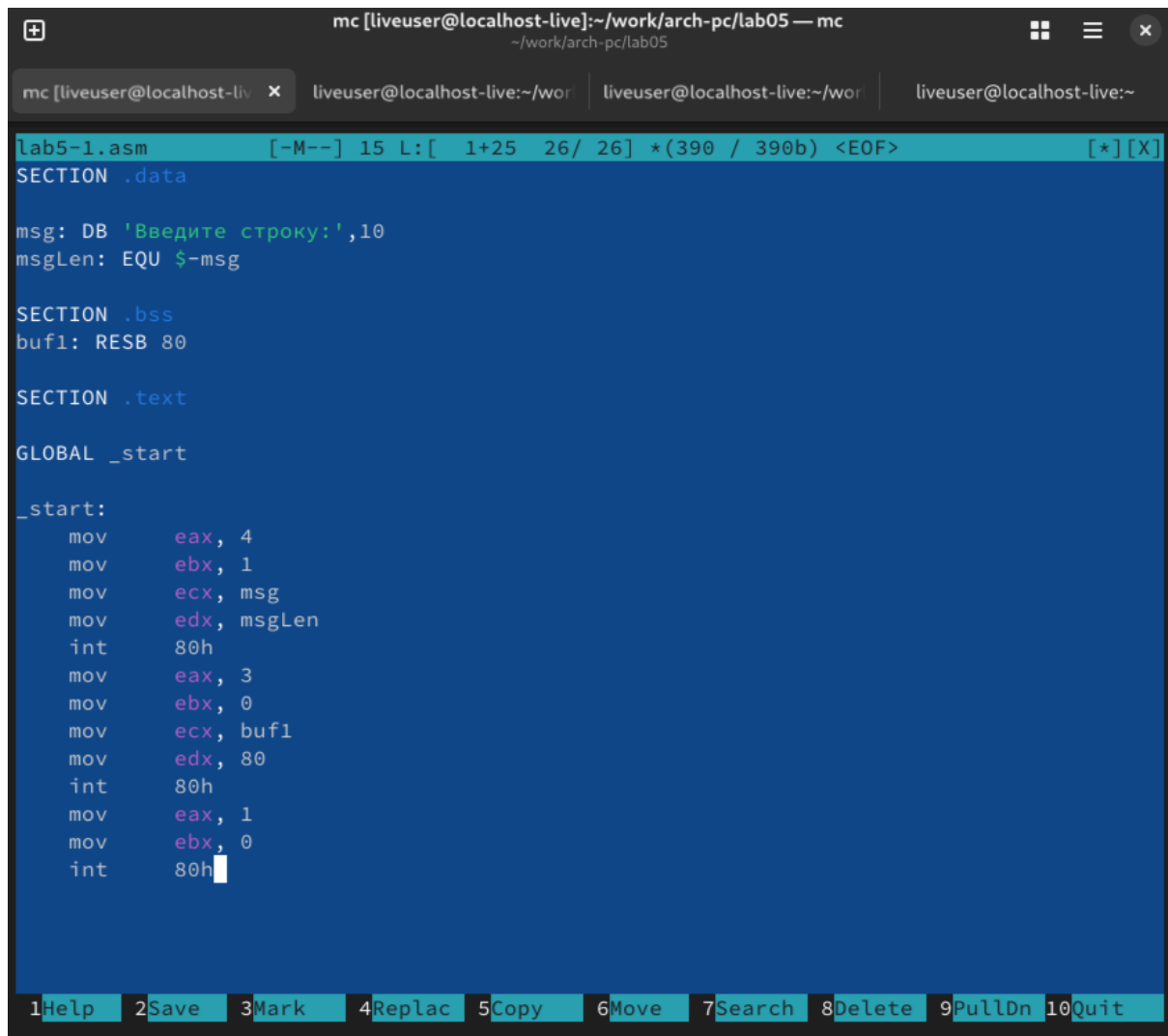
GLOBAL _start

_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
```

The bottom status bar displays a menu of keyboard shortcuts: 1Help, 2Save, 3Mark, 4Replac, 5Copy, 6Move, 7Search, 8Delete, 9PullDn, 10Quit.

Рис. 4.6: Редактирование файла в Midnight Commander

Проверяю сохраненные изменения с помощью клавиши F3 (рис. 4.7).



```
mc [liveuser@localhost-live]:~/work/arch-pc/lab05 — mc
~/work/arch-pc/lab05
lab5-1.asm [-M--] 15 L:[ 1+25 26/ 26] *(390 / 390b) <EOF> [*] [X]
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80

SECTION .text

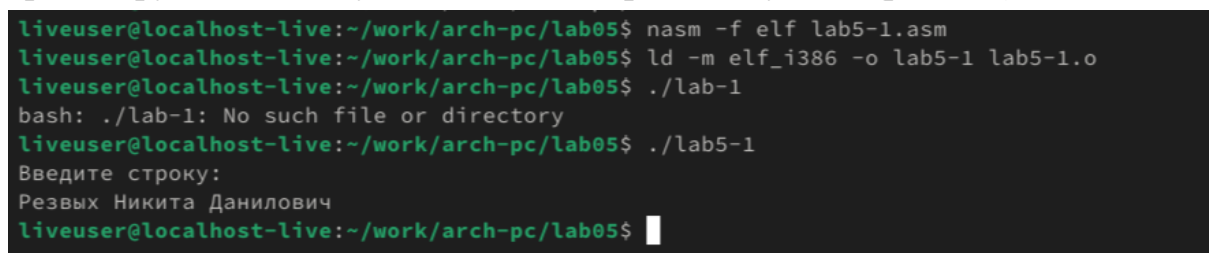
GLOBAL _start

_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
    jmp     _start
    int     80h
```

1Help 2Save 3Mark 4Replac 5Copy 6Move 7Search 8Delete 9PullDn 10Quit

Рис. 4.7: Проверка сохранения сделанных изменений

Транслирую и компоную измененный файл, запускаю (рис. 4.8).



```
liveuser@localhost-live:~/work/arch-pc/lab05$ nasm -f elf lab5-1.asm
liveuser@localhost-live:~/work/arch-pc/lab05$ ld -m elf_i386 -o lab5-1 lab5-1.o
liveuser@localhost-live:~/work/arch-pc/lab05$ ./lab-1
bash: ./lab-1: No such file or directory
liveuser@localhost-live:~/work/arch-pc/lab05$ ./lab5-1
Введите строку:
Резвых Никита Данилович
liveuser@localhost-live:~/work/arch-pc/lab05$
```

Рис. 4.8: Трансляция, компоновка и последующий запуск программы

## **4.3 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. (рис. 4.9).

## **5 Выводы**

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

## **Список литературы**

1. Пример выполнения лабораторной работы
2. Курс на ТУИС
3. Лабораторная работа №5
4. Программирование на языке ассемблера NASM Столяров А. В.