Отчёт по лабораторной работе №5

Дисциплина: Архитектура компьютера

Бережной Иван Александрович

Содержание

# 1 Цель работы

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

# 2 Задание

1. Основы работы с mc
2. Работа с языком ассемблера NASM
3. Подключение внешнего файла
4. Задание для самостоятельной работы

# 3 Выполнение лабораторной работы

## 3.1 Основы работы с mc

Откроем Midnight Commander командой mc и перейдём в каталог ~/work/arch-pc, который мы создали в ходе выполнения лабораторной работы №4 (рис. [1](#fig:001)).

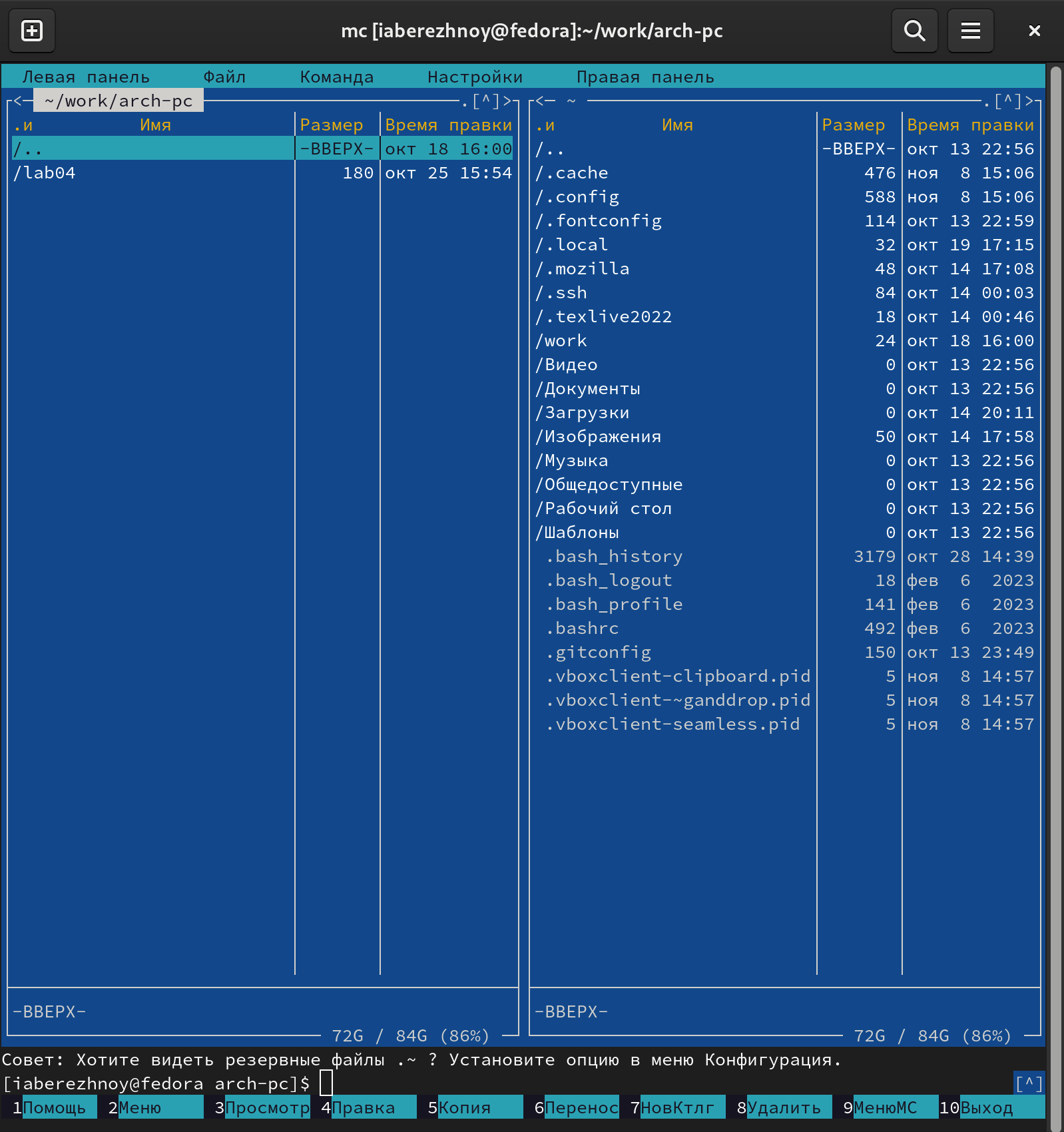


Figure 1: Открытие Midnight Commander

Создадим папку lab05, в которой и будем работать в дальнейшем (рис. [2](#fig:002)). Перейдём в неё.

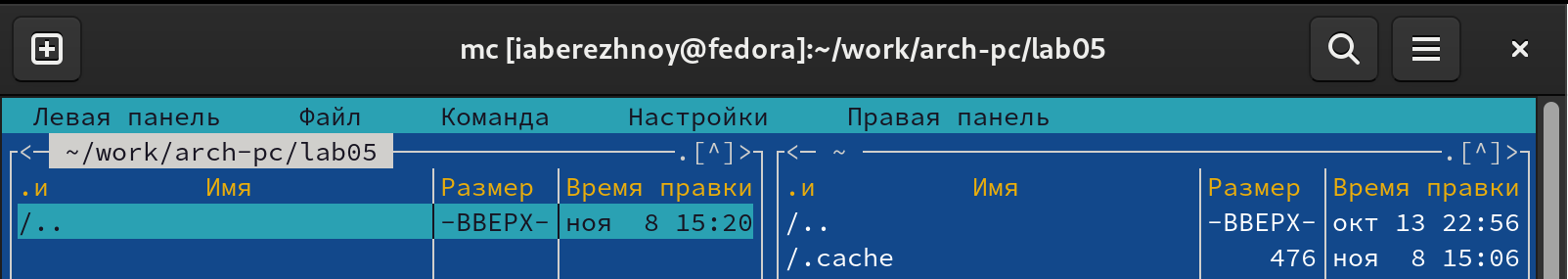


Figure 2: Создание каталога lab05

С помощью команды touch создаём файл lab5-1.asm (рис. [3](#fig:003)).

## 3.2 Работа с языком ассемблера NASM

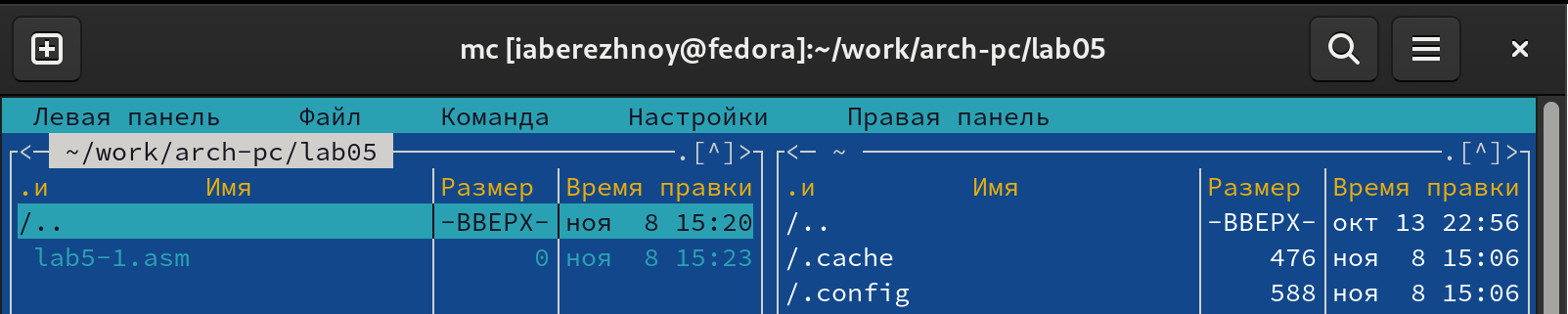


Figure 3: Создание файла lab5-1.asm

Откроем lab5-1.asm, используя функциональную клавишу F4. Мы попали в редактор mcedit. Введём текст программы из предложенного листинга и сохраним изменения (рис. [4](#fig:004)).

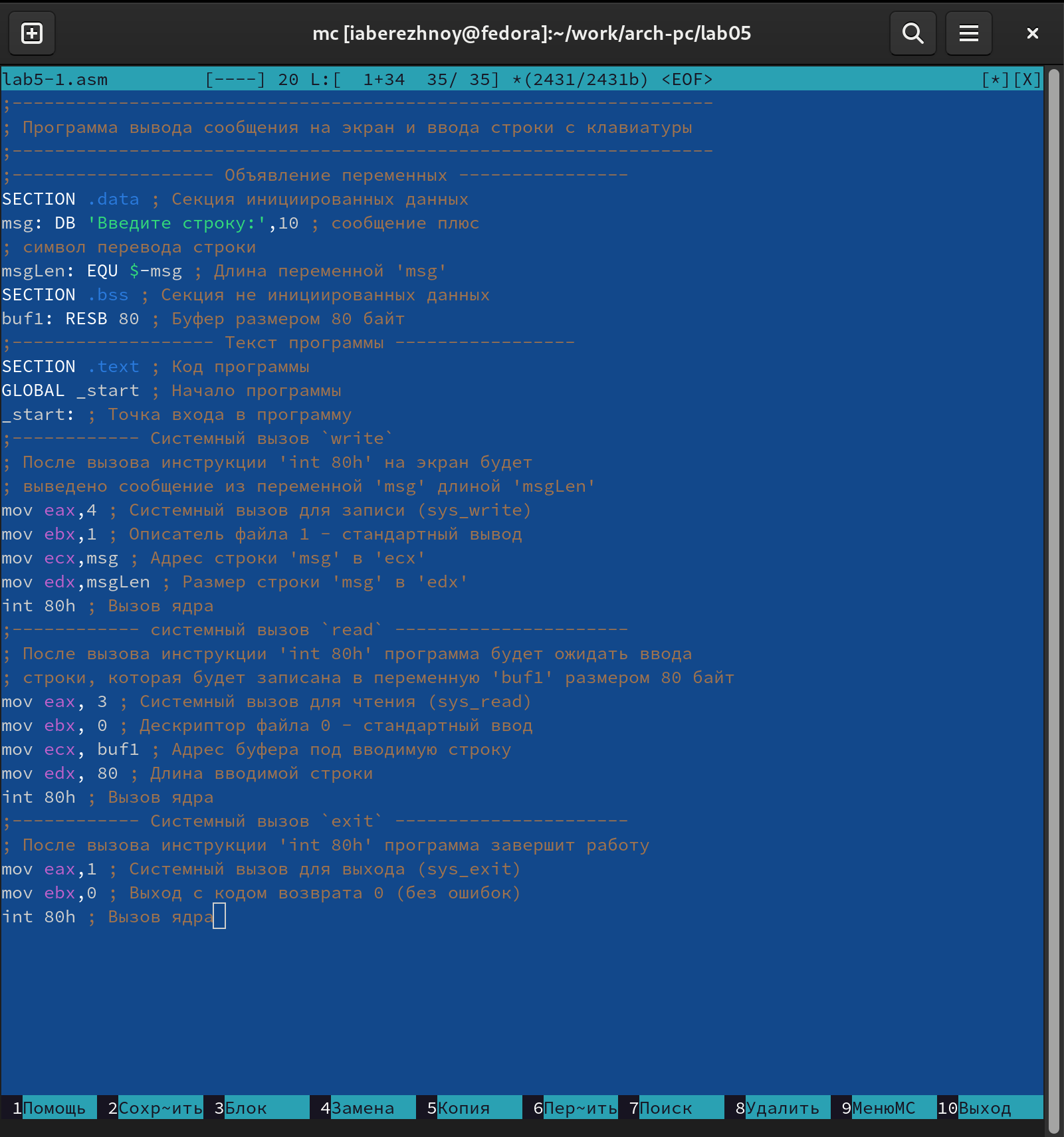


Figure 4: Изменение lab5-1.asm

Откроем файл для просмотра и убедимся, что все изменения сохранились корректно (рис. [5](#fig:005)).

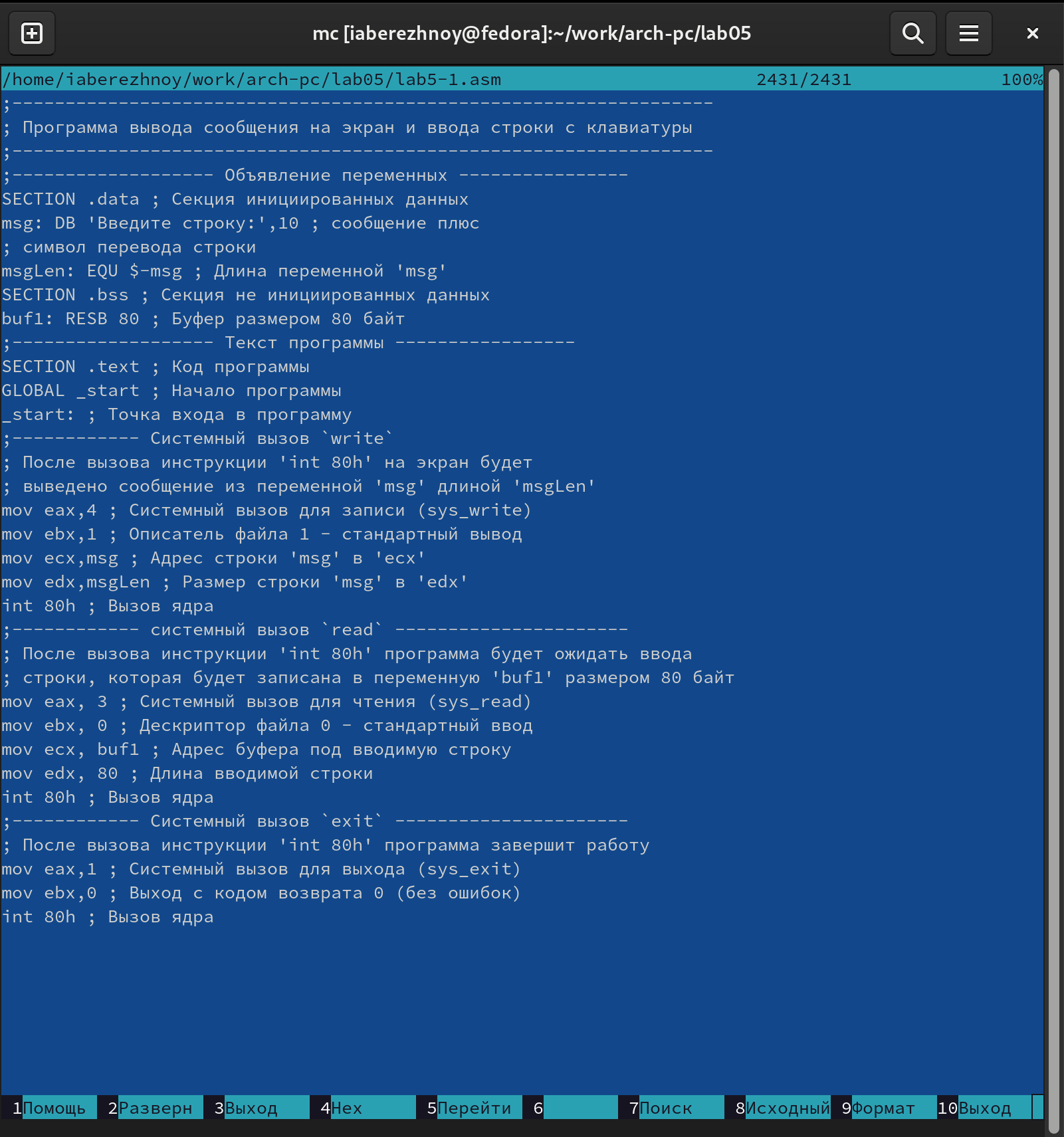


Figure 5: Просмотр lab5-1.asm

Оттранслируем текст программы в объектный файл, затем выполним его компоновку (рис. [6](#fig:006)) и запустим новый исполняемый файл. Подаём на ввод строку “Бережной Иван Александрович” (рис. [7](#fig:007)).

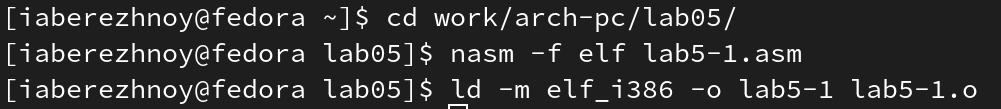


Figure 6: Создание исполняемого файла lab5-1.exe

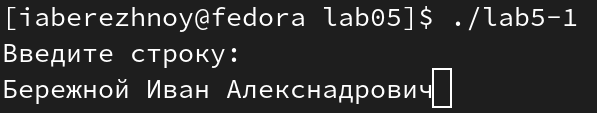


Figure 7: Запуск исполняемого файла lab5-1.exe

## 3.3 Подключение внешнего файла

Скачаем файл in\_out.asm и переместим его в каталог, в котором мы сейчас работаем. Для этого скопируем его из загрузок в ~/work/arch-pc/lab05 (рис. [8](#fig:008)).

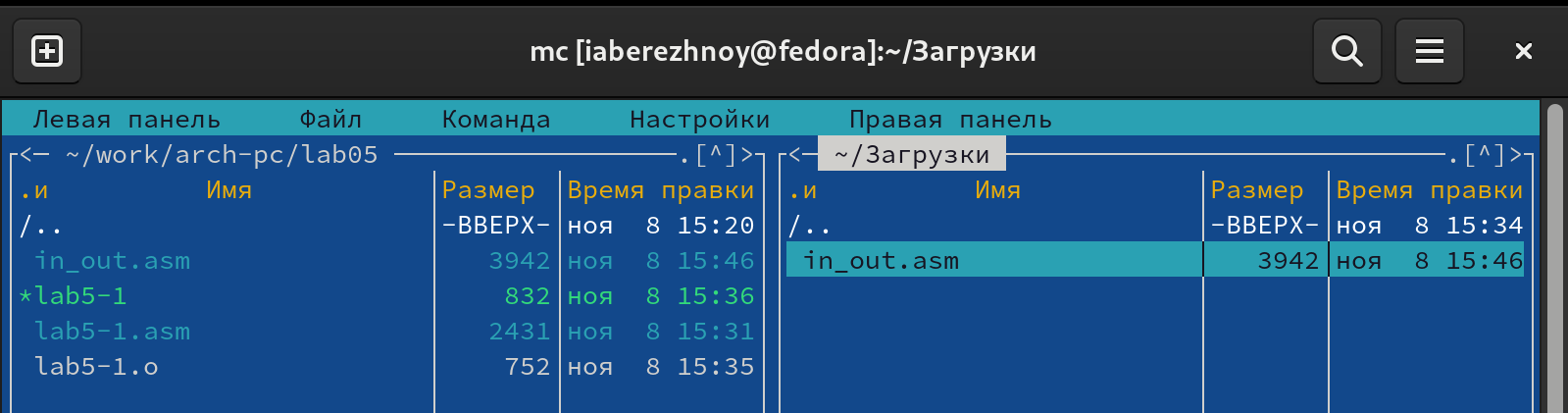


Figure 8: Копирование in\_out.asm

Создадим копию файла lab5-1.asm, назвав lab5-2.asm. Используя предложенный листинг, изменим текст программы в созданном файле с использованием подпрограмм из внешнего файла in\_out.asm (рис. [9](#fig:009)).

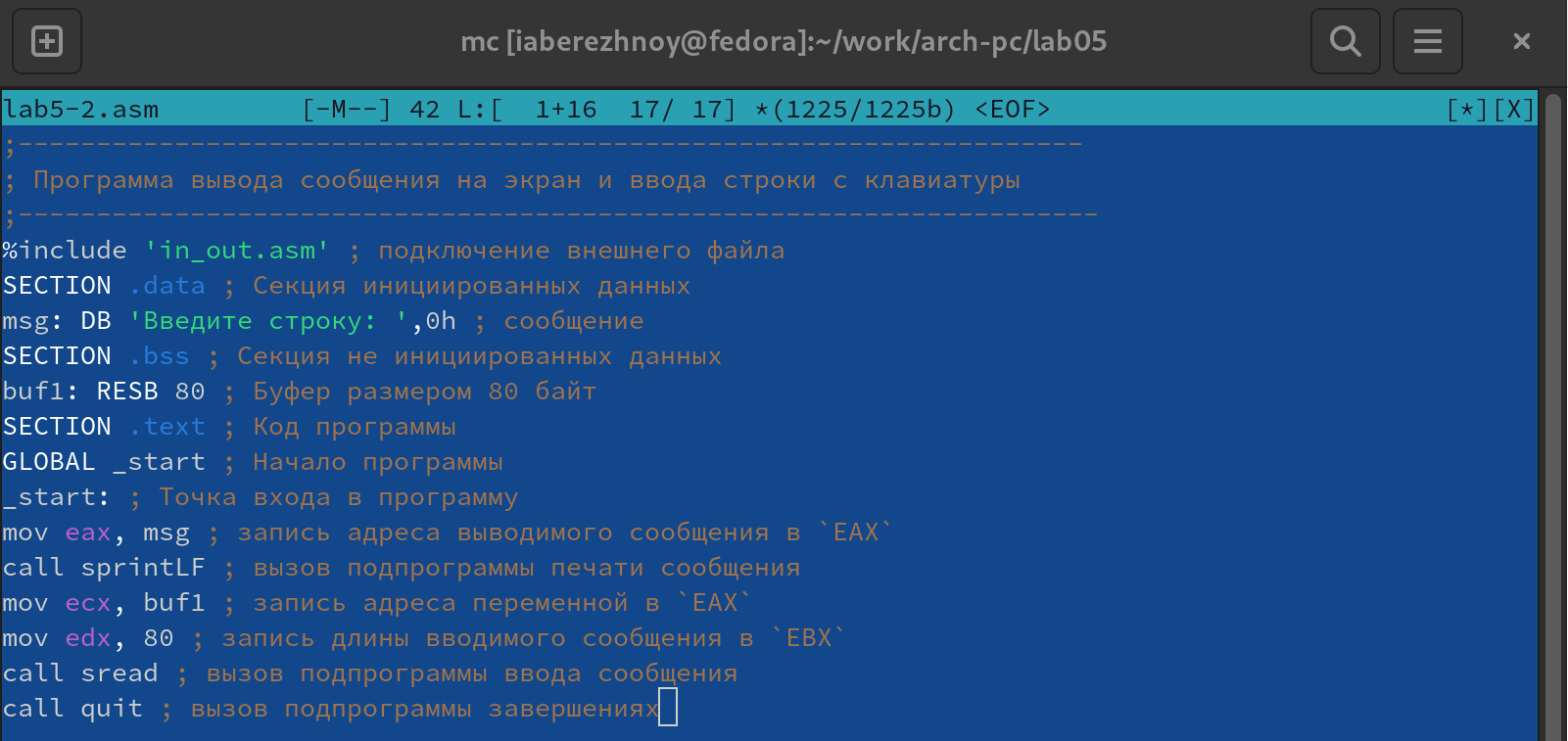


Figure 9: Создание и редактирование lab5-2.asm

Проверим работу файла. Для этого оттранслируем объектный файл, выполним его компоновку и запустим исполняемый файл. Видим запрос на ввод строки. Введём “Бережной Иван Александрович” (рис. [10](#fig:010)).

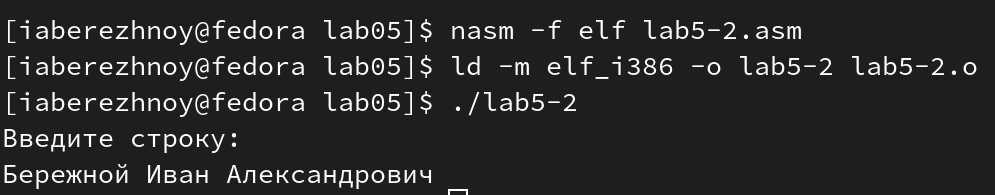


Figure 10: Проверка работы lab5-2.exe

Немного изменим файл lab5-2.asm, заменив sprintLF на sprint. Снова создадим исполняемый файл по той же схеме и запустим его. Видим, что запрос на ввод строки находится на той же строке, что и приглашение типа “Введите строку:” (рис. [11](#fig:011)). В предыдущей версии программы запрос на ввод был на следующей строке после приглашения.

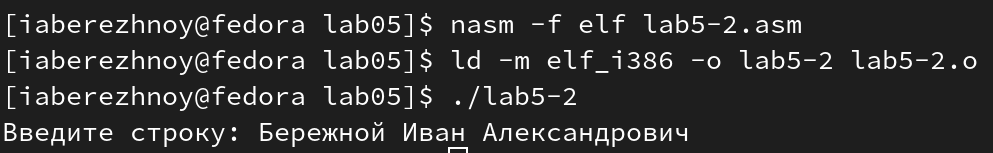


Figure 11: Запуск изменённого файла lab5-2.exe

## 3.4 Задание для самостоятельной работы

Создадим копию файла lab5-1.asm, назвав lab5-3.asm (рис. [12](#fig:012)). Редактируем файл так, чтобы программа выводила вводимую пользователем строку (так же изменим приглашение на “Введите свою фамилию:”) (рис. [13](#fig:013)).

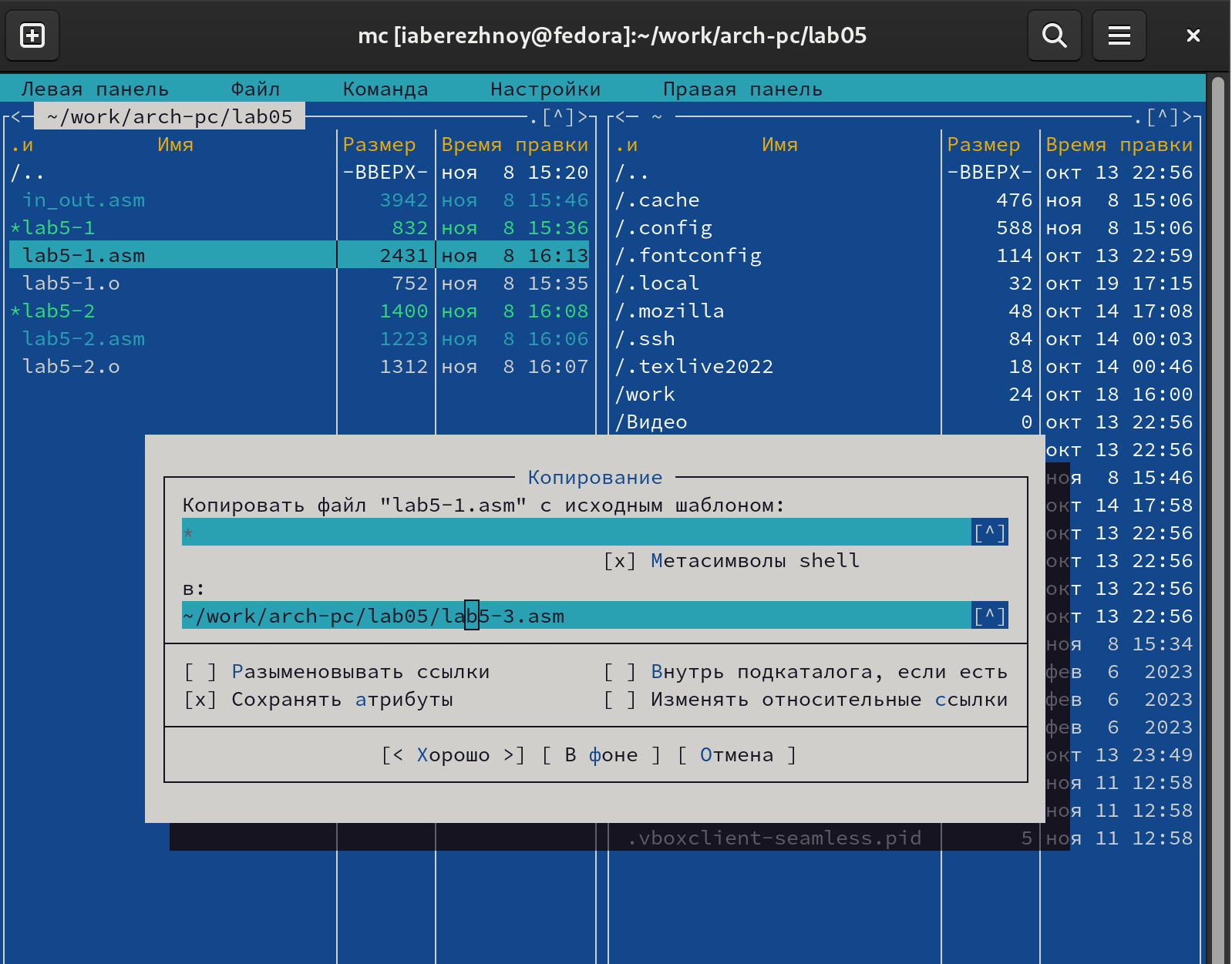


Figure 12: Создание файла lab5-3.asm

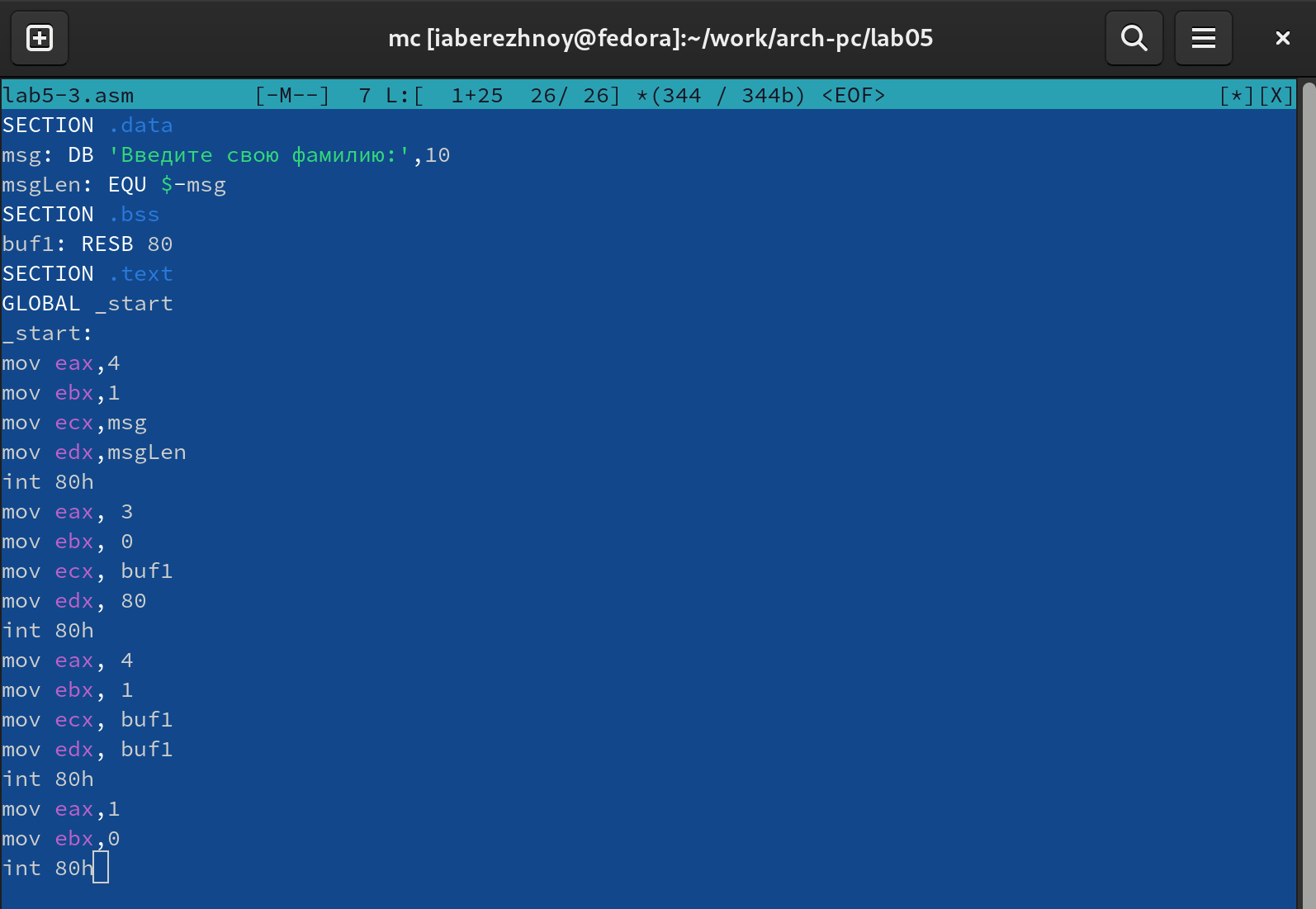


Figure 13: Редактирование lab5-3.asm

Теперь создадим исполняемый файл по знакомой нам схеме и запустим его (рис. [14](#fig:014)). Всё работает корректно.

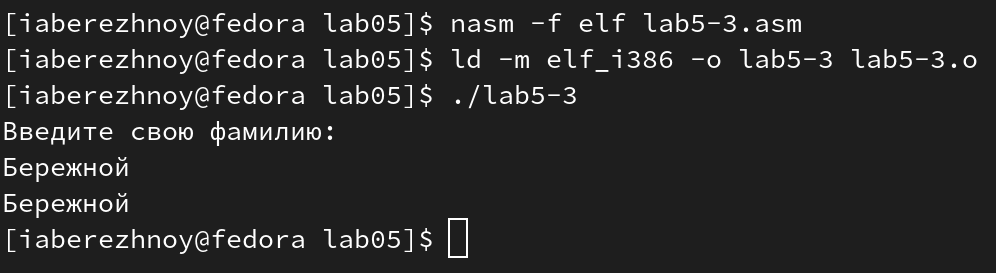


Figure 14: Запуск lab5-3.exe

**Листинг 5.1**

SECTION .data  
msg: DB 'Введите свою фамилию:',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

Создадим копию файла lab5-2.asm, назвав lab5-4.asm (рис. [15](#fig:015)). Редактируем файл так, чтобы он выполнял те же функции, что и lab5-3.asm (рис. [16](#fig:016)).

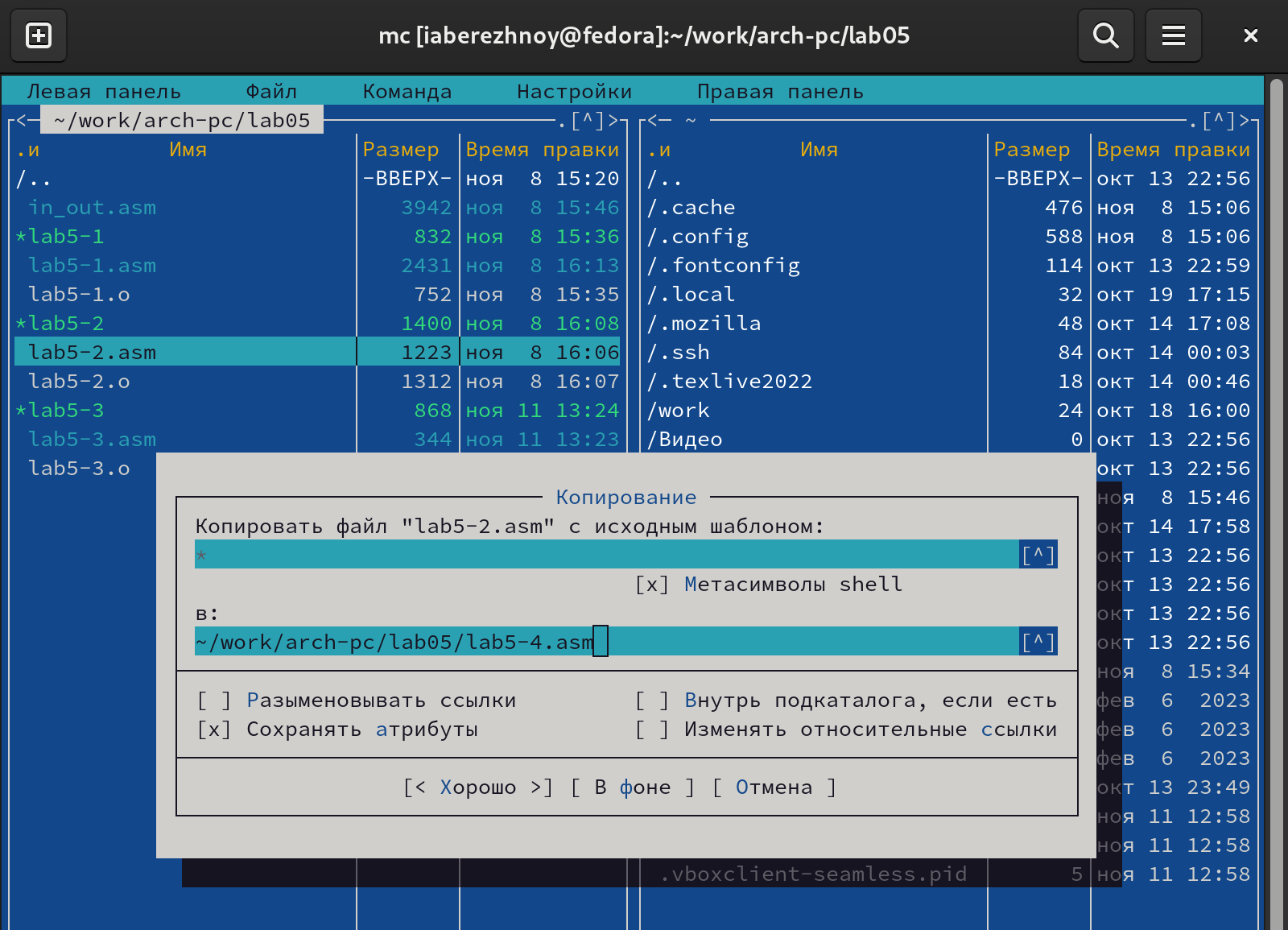


Figure 15: Создание файла lab5-4.asm

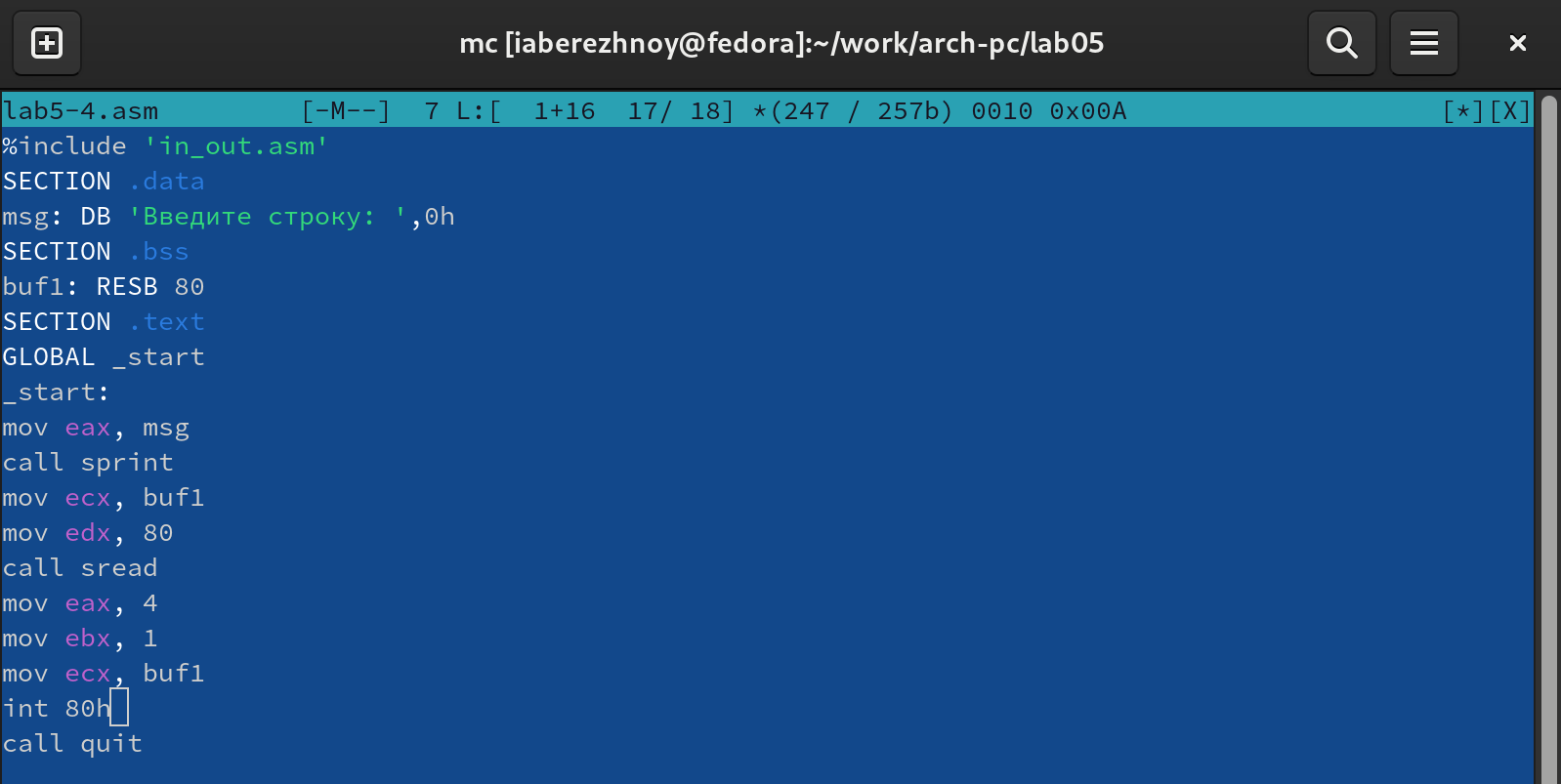


Figure 16: Редактирование lab5-4.asm

Теперь создадим исполняемый файл по знакомой нам схеме и запустим его (рис. [17](#fig:017)). Всё работает корректно.

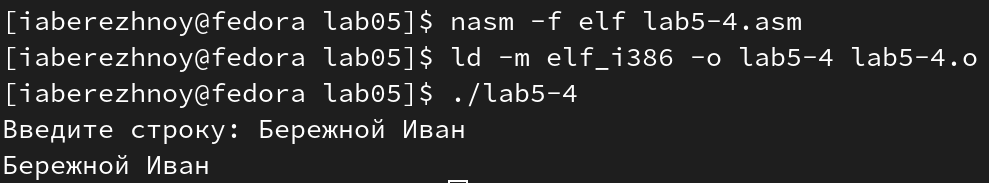


Figure 17: Запуск lab5-4.exe

**Листинг 5.2**

%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

# 4 Вывод

В ходе выполнения лабораторной работы мы приобрели практические навыки работы в Midnight Commander и освоили инструкции языка ассемблера mov и int.

# Список литературы

::: [Архитектура ЭВМ](https://esystem.rudn.ru/mod/resource/view.php?id=1030553)