Отчёт по лабораторной работе №5

дисциплина: Архитектура компьютера

Самойлова Софья Дмитриевна

Содержание

# 1 Цель работы

Целью 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. Подключение внешнего файла in\_out.asm
3. 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 инициированных (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 инициированных 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 размеров 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 intпредназначена 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 mc

Открываю *Midnight Commander* и перехожу в каталог ~/work/arch-pc созданный при выполнении лабораторной работы №4 (рис. 1).

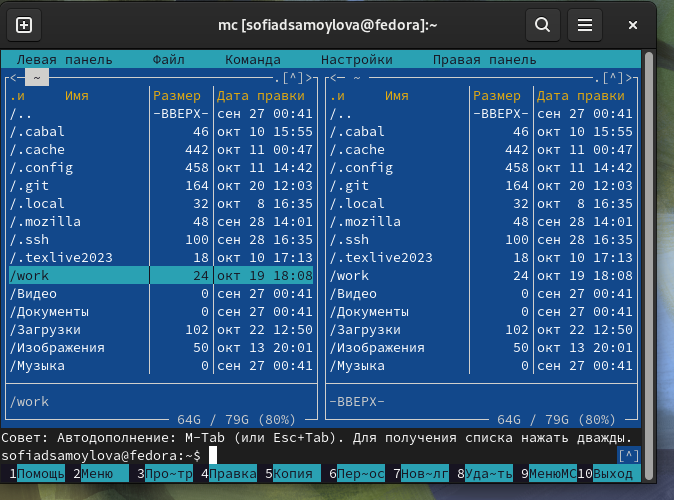


Рис. 1: Midnight Commander

С помощью функциональной клавиши F7 создаю папку lab05 и перехожу в созданный каталог (рис. 2).

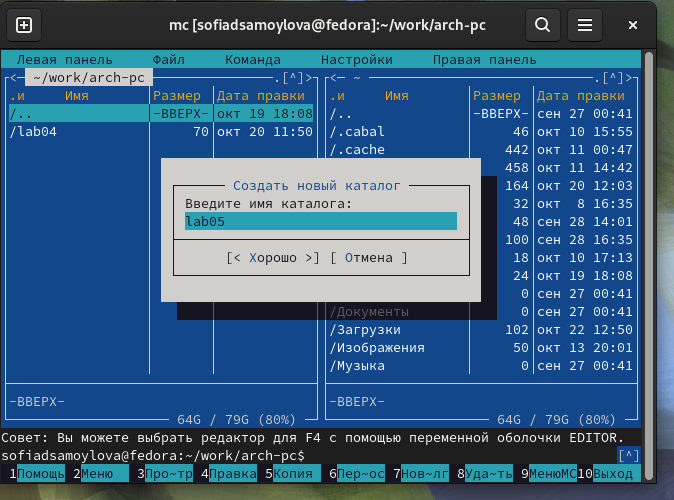


Рис. 2: Создание lab05

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

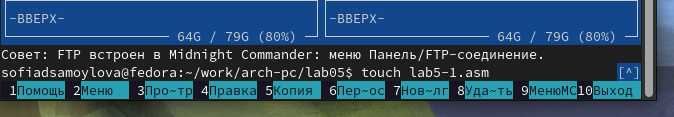


Рис. 3: Создание lab05-1.asm

С помощью функциональной клавиши F4 открываю файл lab5-1.asm для редактирования во встроенном редакторе. В качестве встроенного редактора *Midnight Commander* используется редактор mcedit. Ввожу текст программы из листинга, сохраняю изменения. (рис. 4).

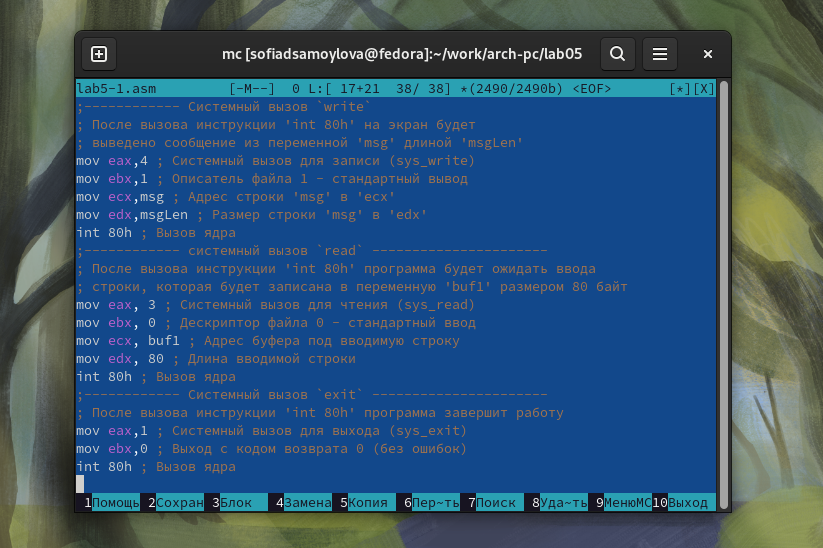


Рис. 4: Редактирование lab05-1.asm

Оттранслирую 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. 5).

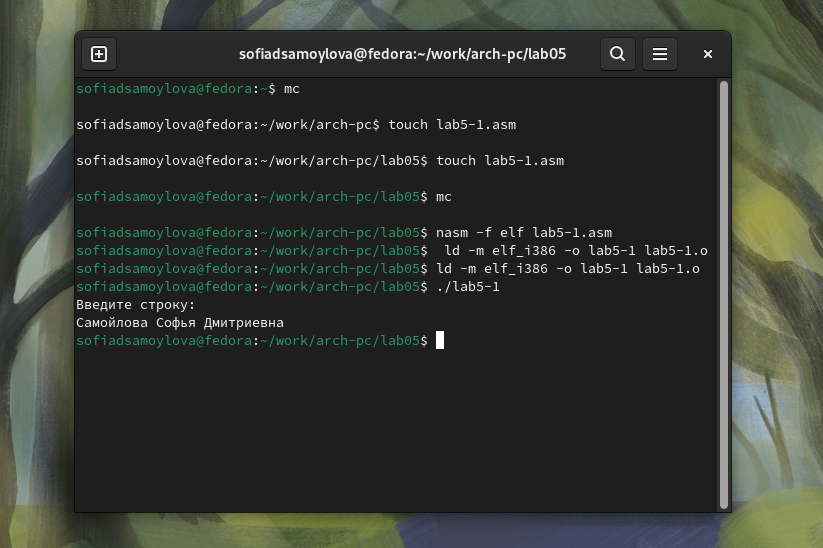


Рис. 5: Работа файла

## 4.2 Подключение внешнего файла

Скачиваю файл in\_out.asm со страницы курса в ТУИС (рис. 6).

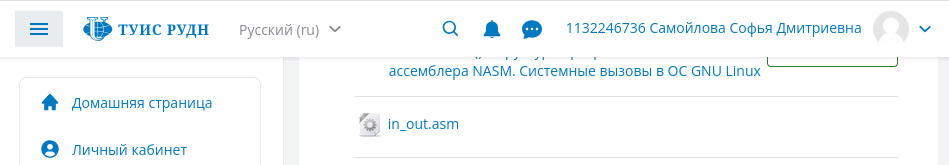


Рис. 6: Файл на ТУИС

Копирую файл in\_out.asm в каталог с файлом lab5-1.asm с помощью функциональной клавиши F5 (рис. 7).

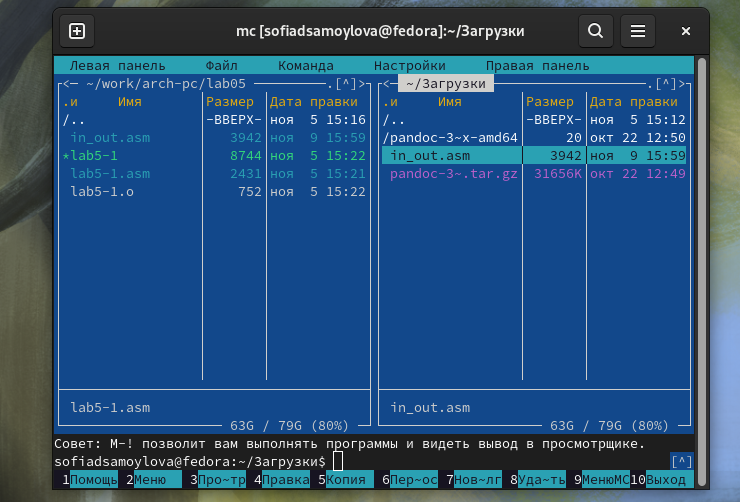


Рис. 7: Копирование in\_out.asm

С помощью функциональной клавиши F6 создаю копию файла lab5-1.asm с именем lab5-2.asm. Выделяю файл lab5-1.asm, нажмимаю клавишу F6 , ввожу имя файла lab5-2.asm и нажимаю клавишу Enter (рис. 8).

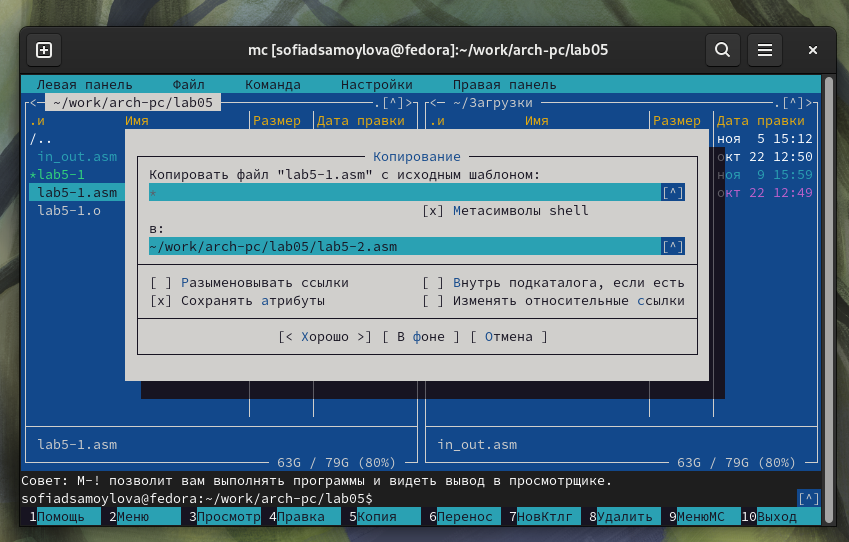


Рис. 8: Создание копии

Исправляю текст программы в файле lab5-2.asm с использование подпрограмм из внешнего файла in\_out.asm. Создаю исполняемый файл и проверяю его работу (рис. 9).

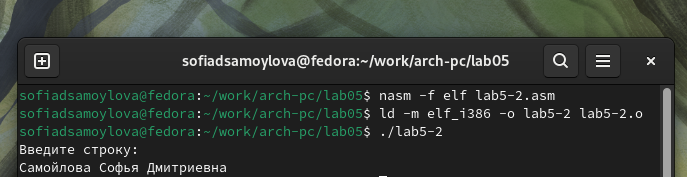


Рис. 9: Работа нового файла

В файле lab5-2.asm заменяю подпрограмму sprintLF на sprint(рис. 10).

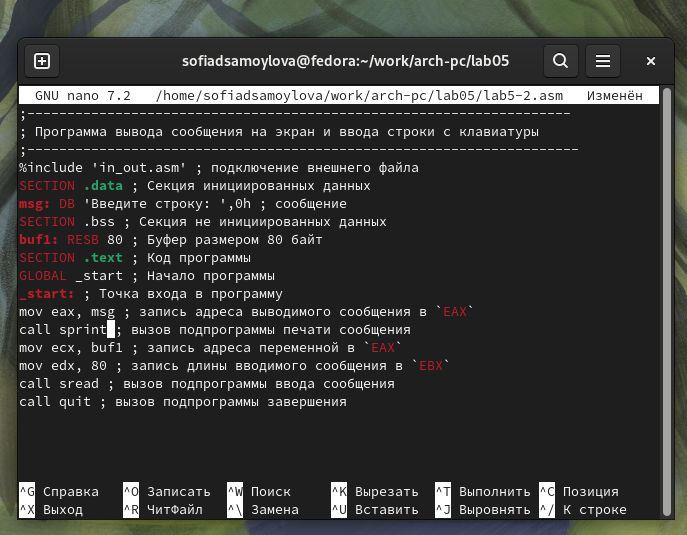


Рис. 10: Внесение изменений

Создаю исполняемый файл и проверяю его работу (рис. 11).

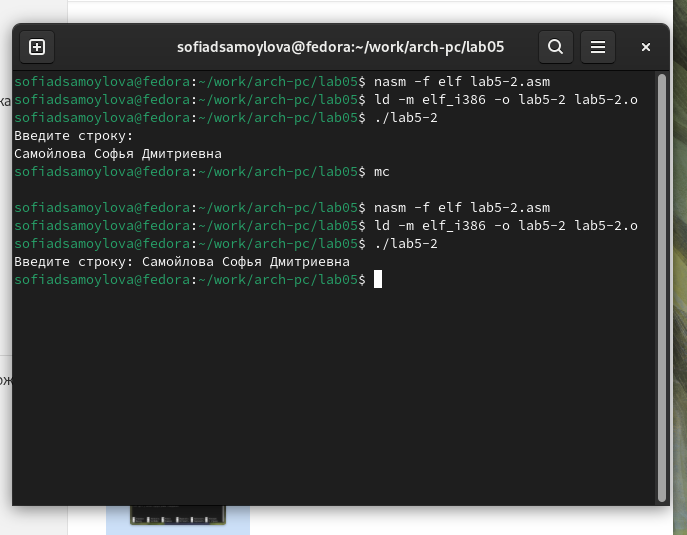


Рис. 11: Работа файла после исправлений

Разница между исполняемыми файлами в том, что запуск первого запрашивает ввод с новой строки, а программа, которая исполняется при запуске второго, запрашивает ввод без переноса на новую строку, в этом и заключается само различие между подпрограммами sprintLF и sprint.

## 4.3 Выполнение самостоятельной работы

Создаю копию файла lab5-1.asm(рис. 12).

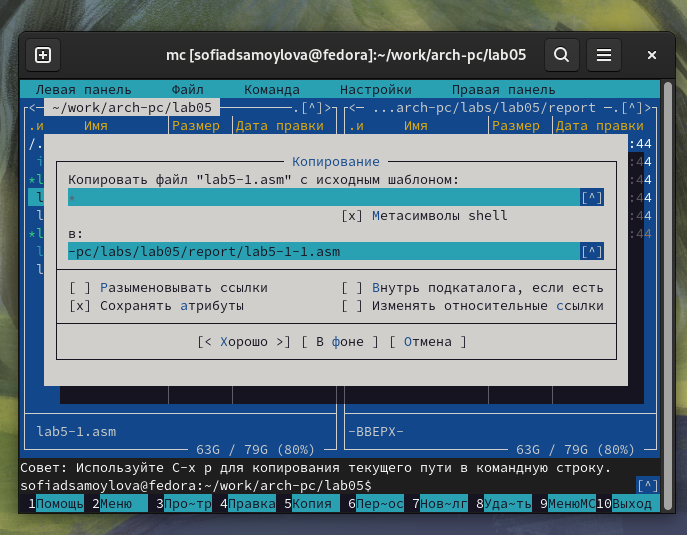


Рис. 12: Создание копии

Вношу изменения в программу, так чтобы она работала по следующему алгоритму (рис. 13): • вывести приглашение типа “Введите строку:”; • ввести строку с клавиатуры; • вывести введённую строку на экран.

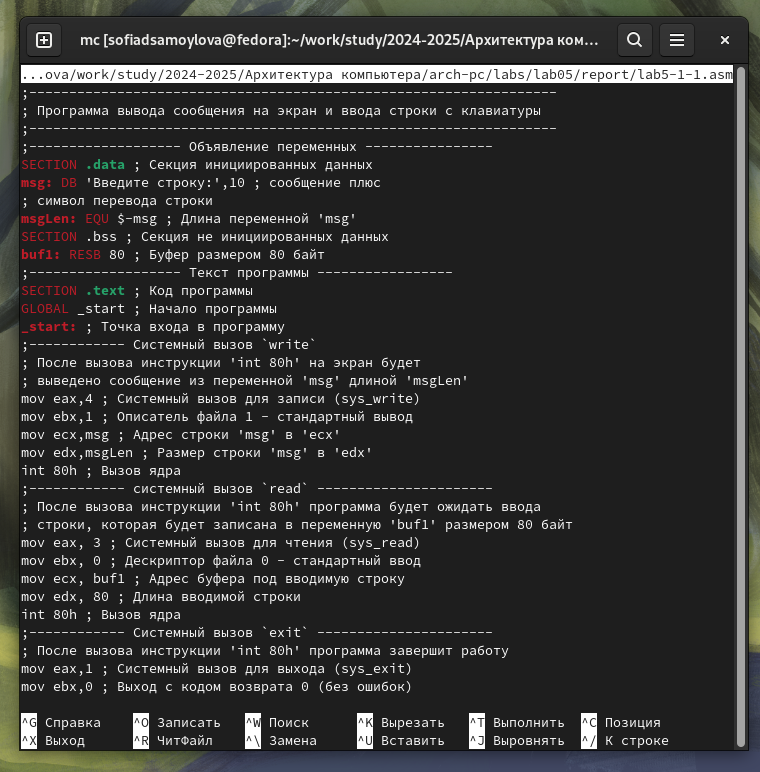


Рис. 13: Работа в копии

Получаю исполняемый файл и проверяю его работу. На приглашение ввести строку ввожу свои ФИО (рис. 14).

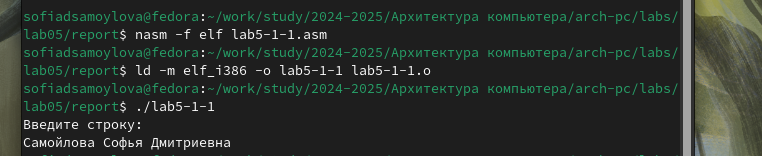


Рис. 14: Работа копии

Создаю копию файла lab5-2.asm. Исправляю текст программы с использование подпрограмм из внешнего файла in\_out.asm (рис. 15), так чтобы она работала по следующему алгоритму: • вывести приглашение типа “Введите строку:”; • ввести строку с клавиатуры; • вывести введённую строку на экран.

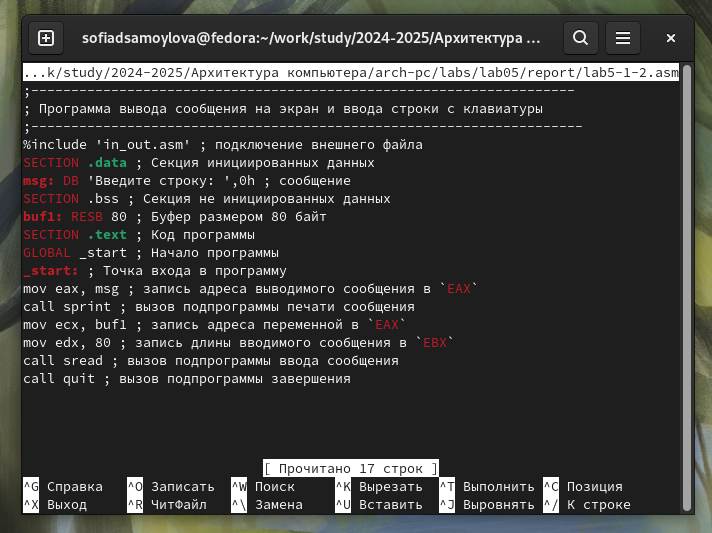


Рис. 15: Редактирование копии

Создаю исполняемый файл и проверяю его работу (рис. 16).

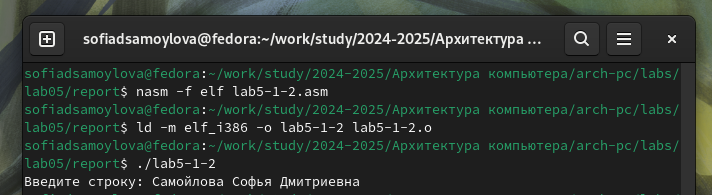


Рис. 16: Работа копии

# 5 Выводы

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

# 6 Список литературы

1. [Лабораторная работа №5](https://esystem.rudn.ru/pluginfile.php/2089085/mod_resource/content/0/%D0%9B%D0%B0%D0%B1%D0%BE%D1%80%D0%B0%D1%82%D0%BE%D1%80%D0%BD%D0%B0%D1%8F%20%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%B0%20%E2%84%965.%20%D0%9E%D1%81%D0%BD%D0%BE%D0%B2%D1%8B%20%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D1%8B%20%D1%81%20Midnight%20Commander%20%28%29.%20%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0%20%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D1%8B%20%D0%BD%D0%B0%20%D1%8F%D0%B7%D1%8B%D0%BA%D0%B5%20%D0%B0%D1%81%D1%81%D0%B5%D0%BC%D0%B1%D0%BB%D0%B5%D1%80%D0%B0%20NASM.%20%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%BD%D1%8B%D0%B5%20%D0%B2%D1%8B%D0%B7%D0%BE%D0%B2%D1%8B%20%D0%B2%20%D0%9E%D0%A1%20GNU%20Linux.pdf)