Лабораторная работа номер 5

Отчёт

Виноградова Мария Андреевна

Содержание

# 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 инициированных (известных во время компиляции) данных (SECTION .data) и секция не инициализированных данных(тех, под которые во время компиляци и только отводится память, а значение присваивается входе выполнения программы)(SECTION.bss).Для объявления инициированных данных в секции .data используются директивы DB, DW,DD,DQиDT, которые резервируют память и указывают,какие значения должны храниться в этой памяти:-DB(define byte)—определяет переменную размером в 1байт;-DW(define word)—определяет переменную размеров в 2 бай та (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.movdst,src Здесь операнд dst—приёмник,аsrc—источ ник. 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 Порядок выполнения лабораторной работы

Открываем Midnight Commander (рис. 1).

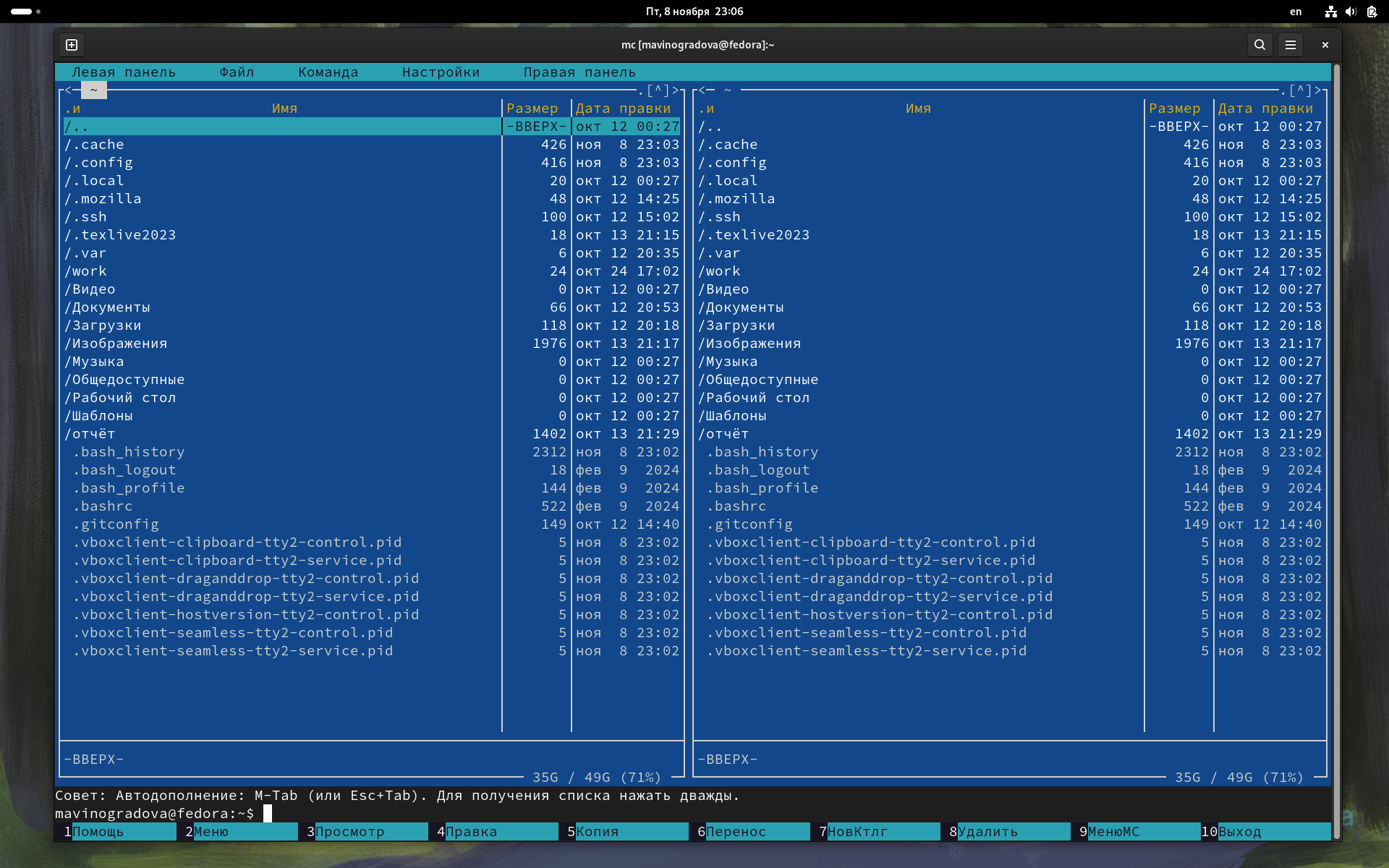


Рис. 1: Вводим в консоль команду mc

Переходим в каталог, созданный при выполнении 4 ЛБ (рис. 2).

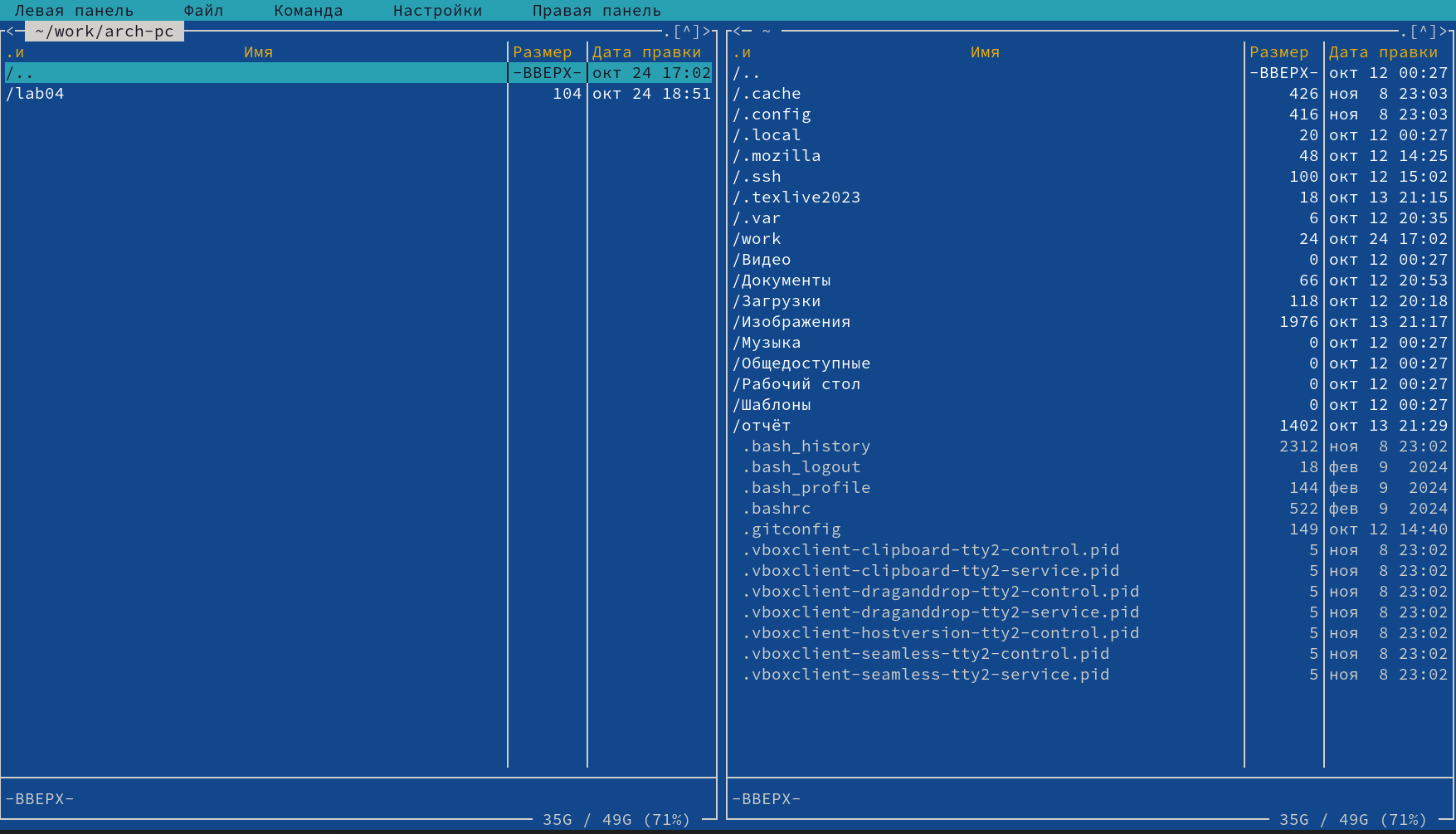


Рис. 2: Переходим в каталог

Создаем каталог lab05 (рис. 3).

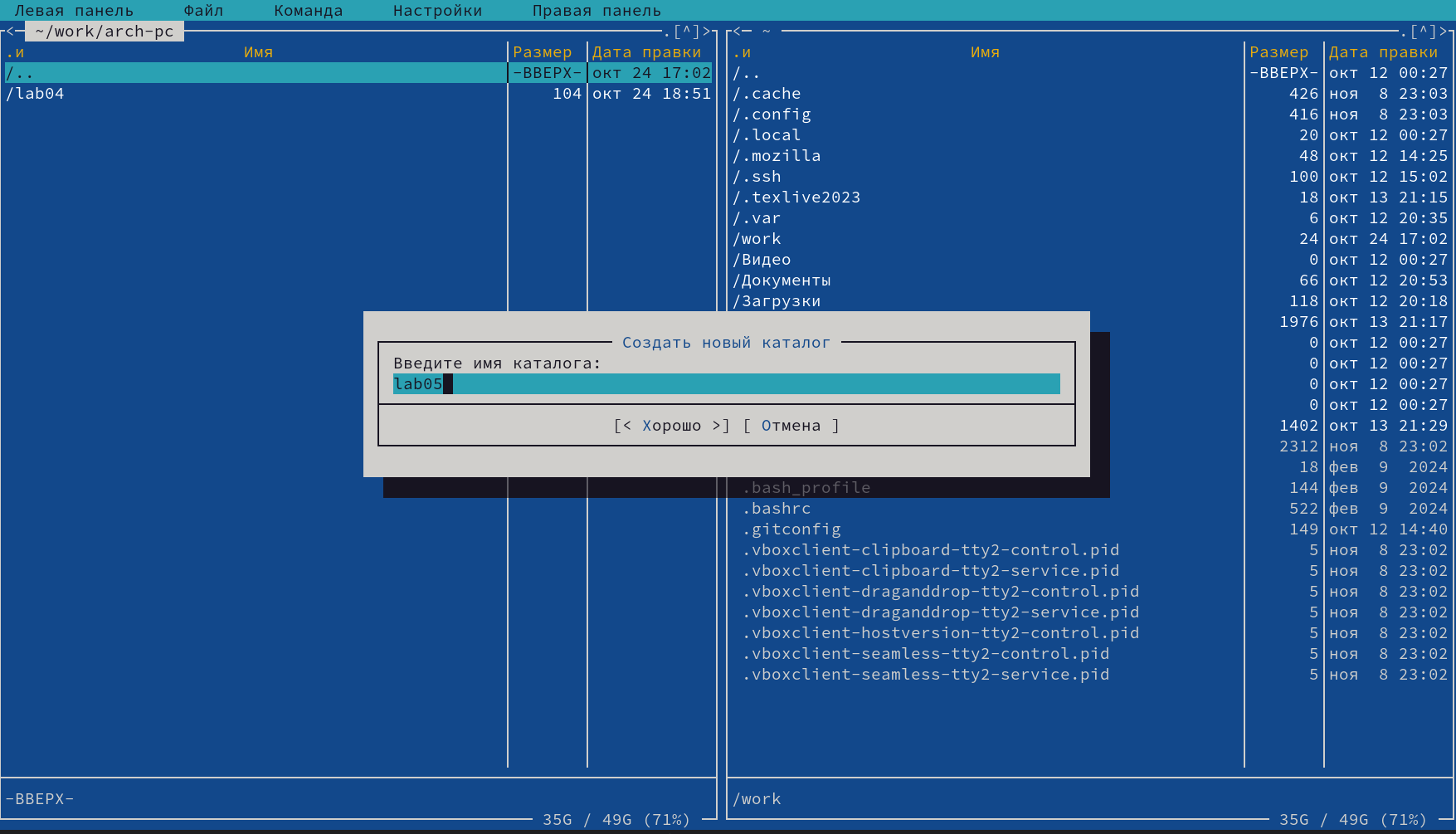


Рис. 3: Создаем каталог функциональной клавишей F7

Создаем файл lab5-1.asm (рис. 4).

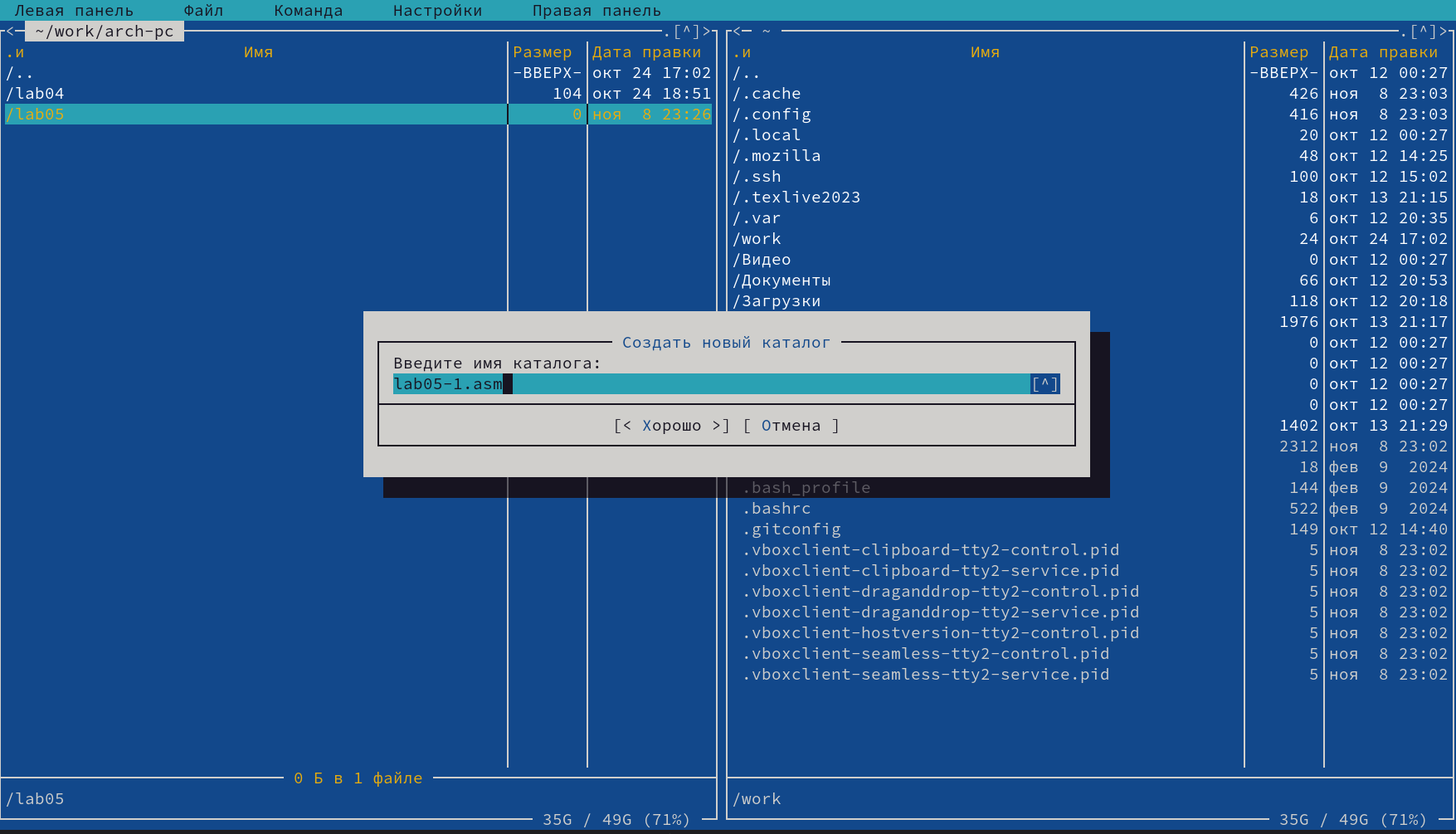


Рис. 4: Воспользуемся командой touch

Открываем файл для редактирования и заполняем его по листингу (рис. 5).

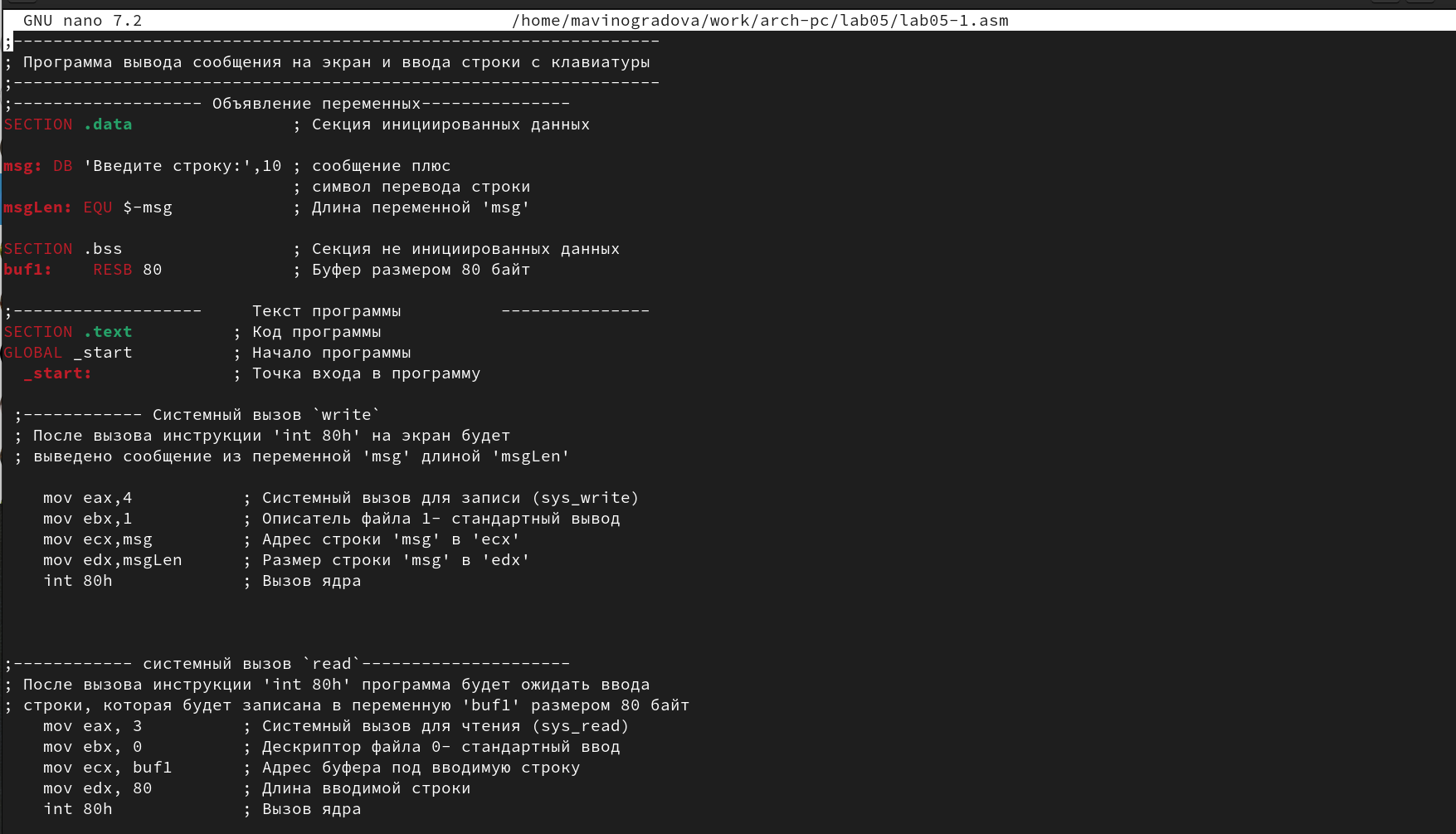


Рис. 5: Открывем файл функциональной клавишей, заполняем и сохраняем

Открывем файл для просмотра (рис. 6).

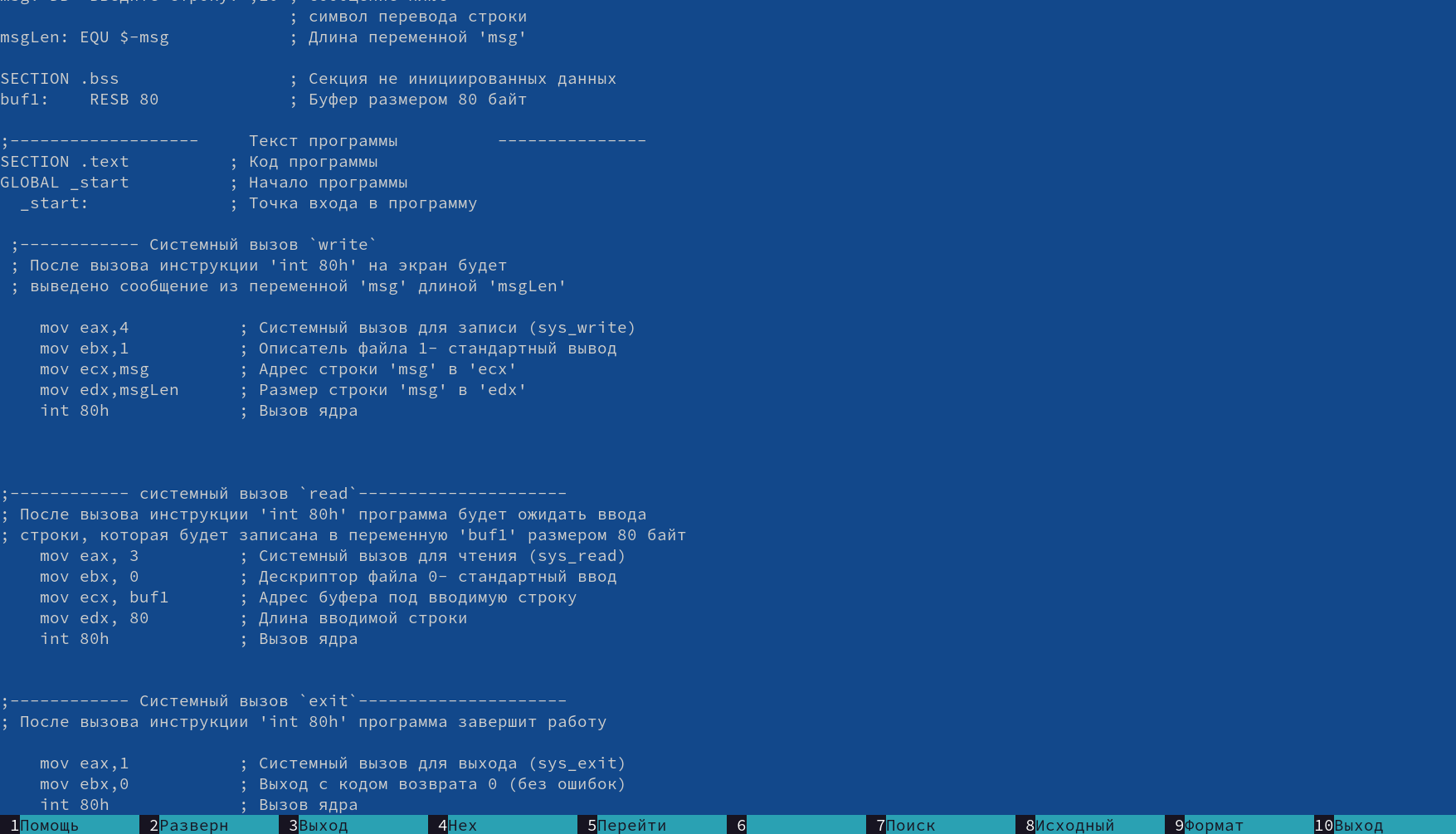


Рис. 6: Открываем файл и убеждаемся, что файл содержит текст программы

Транслируем текст программы и запускаем исполняемый файл (рис. 7).

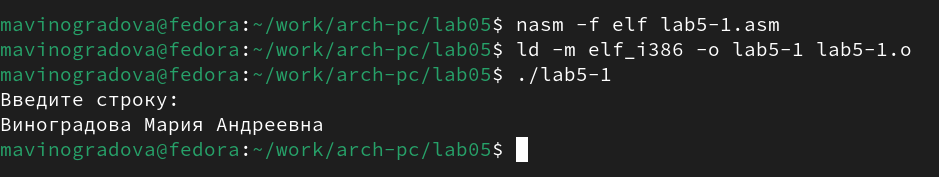


Рис. 7: Проверяем, как работает данная программа

Скачиваем файл со страницы курса (рис. 8).

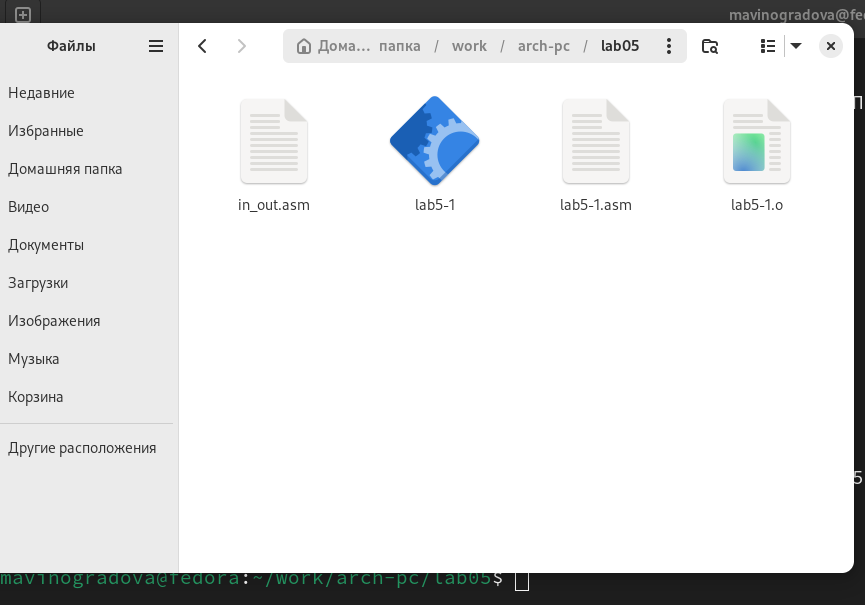


Рис. 8: Скачиваем файл

Копируем файл в нужную директорию (рис. 9).

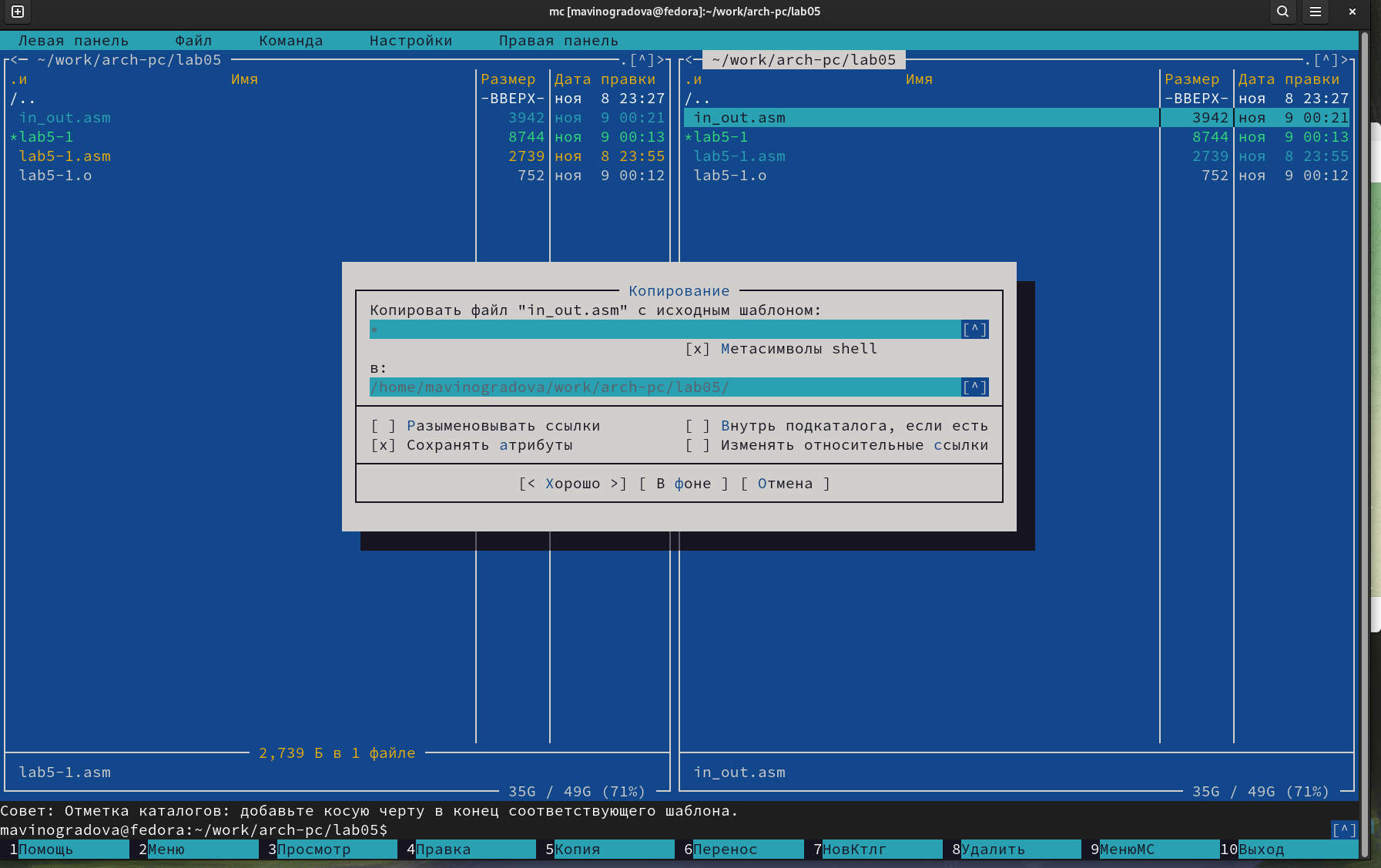


Рис. 9: Копируем скаченный файл

Создаем копию файла lab5-1.asm (рис. 10).

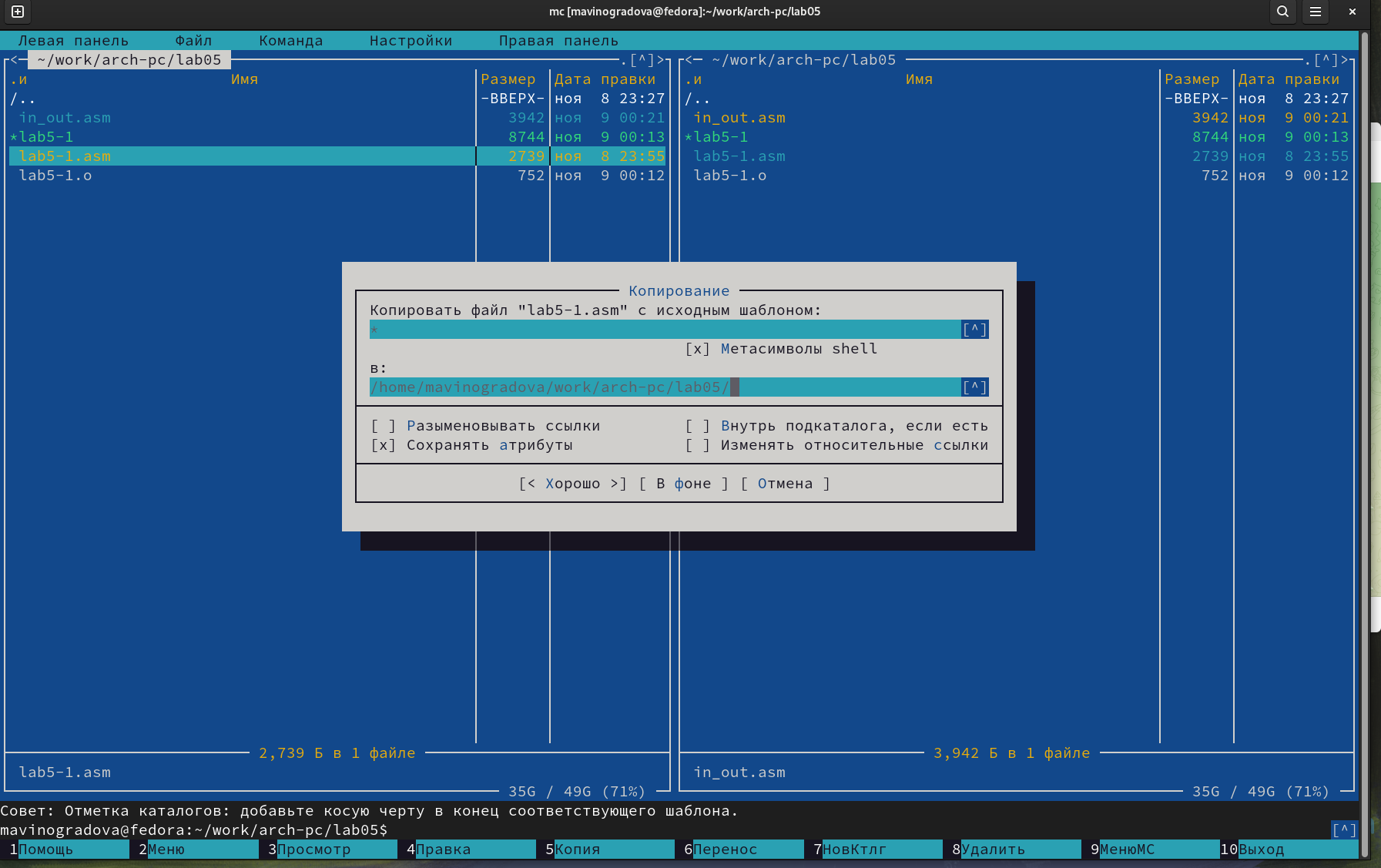


Рис. 10: Создаем копию файла клавишей F6

Проверяем созданный файл (рис. 11).

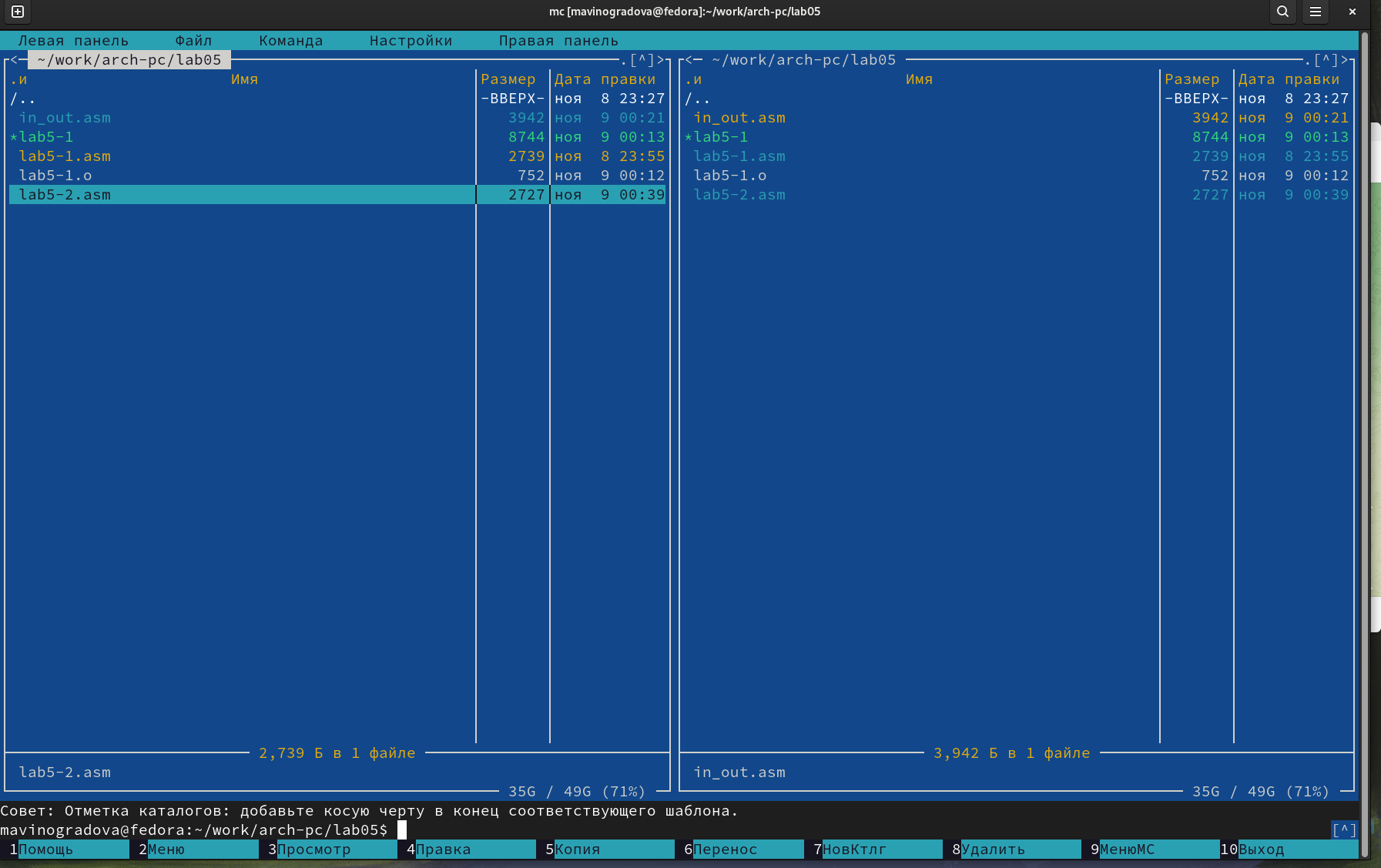


Рис. 11: Проверяем скопировался ли файл

Открываем новый файл и заполняем его в соответствии с листингом (рис. 12).

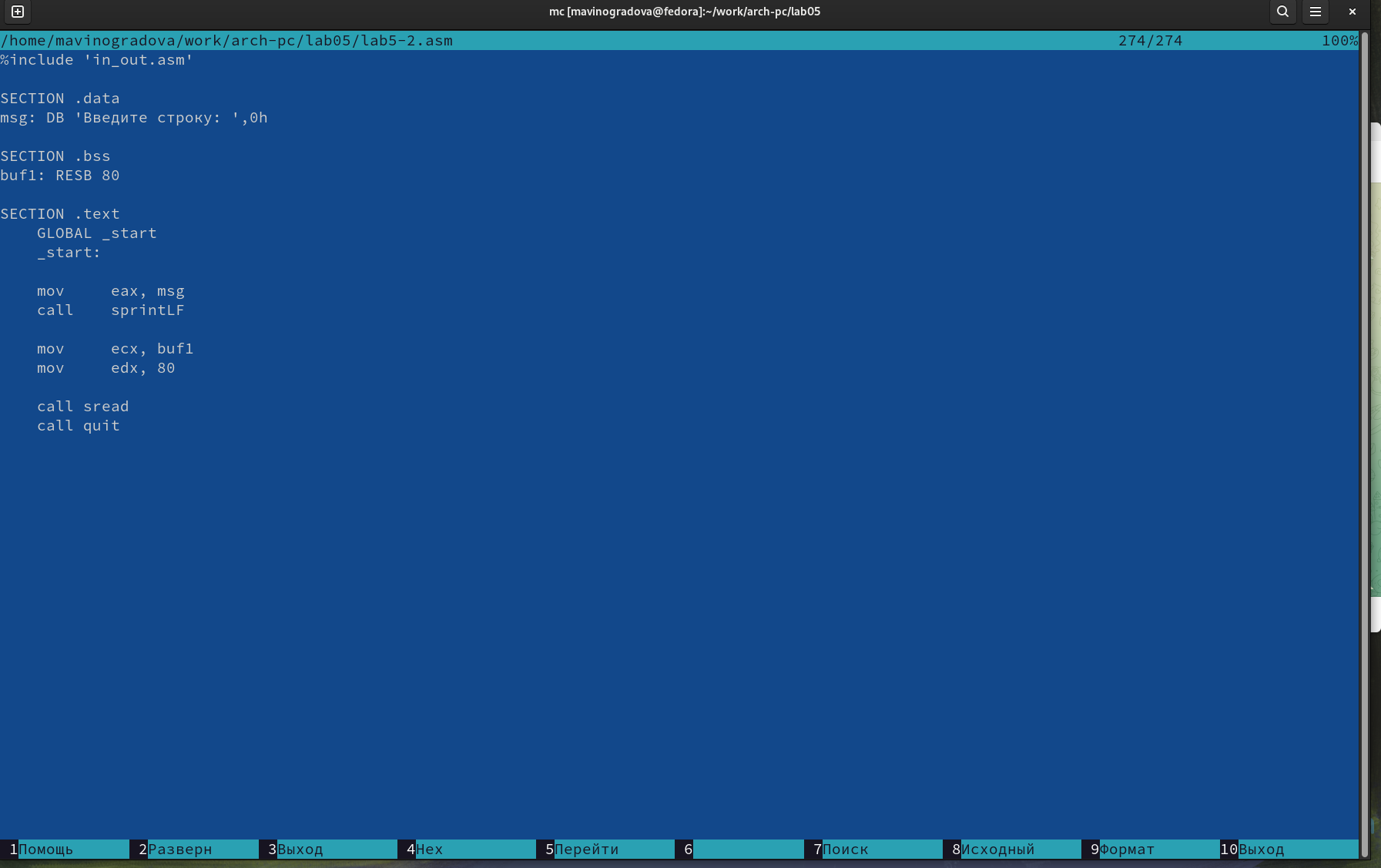


Рис. 12: Открываем и заполняем файл

Транслируем и запускаем новый файл (рис. 13).

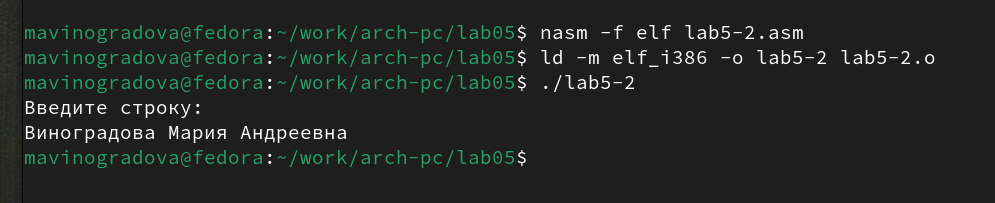


Рис. 13: Смотрим, как сработала программа

Снова открываем файл для редактирования и меняем sprintLF на sprint (рис. 14).

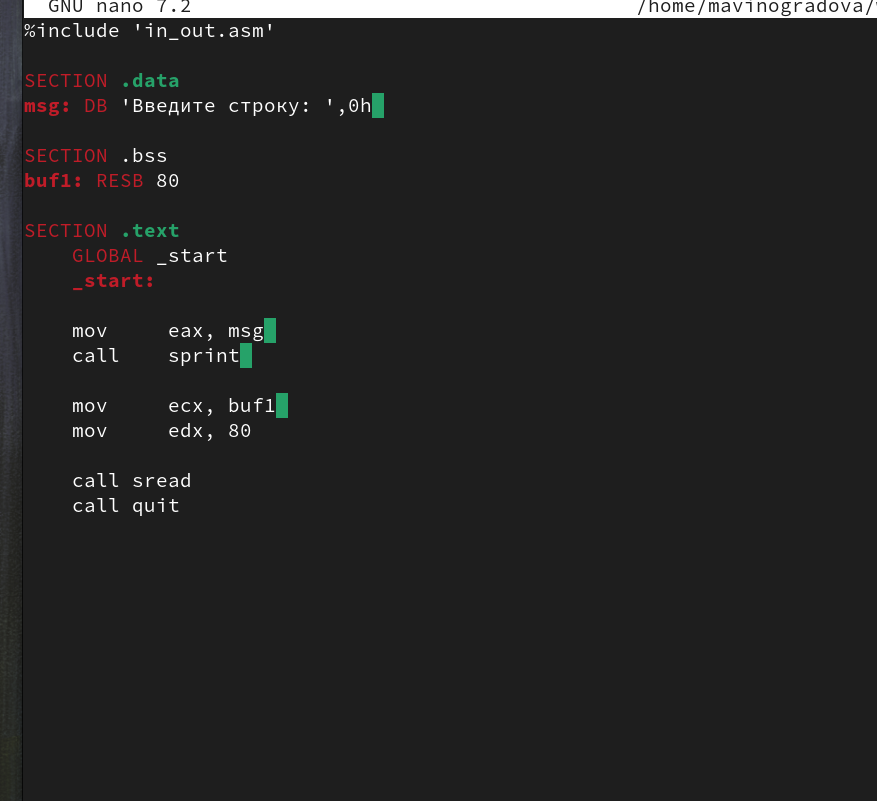


Рис. 14: Редактируем файл

Транслируем и запускаем файл (рис. 15).

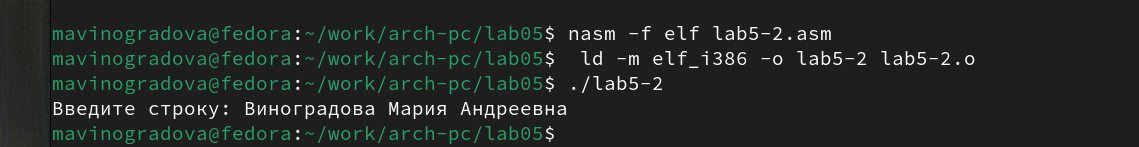


Рис. 15: Смотрим, как сработал программа и сравниваем с прошлой

Таким образом можем понять, что команда sprint выводит текст в той же строке, а sprintLF переносит на новую строку.

## 4.2 Задание для самостоятельной работы

Создаем копию файла lab5-1.asm и называем его так же (рис. 16).

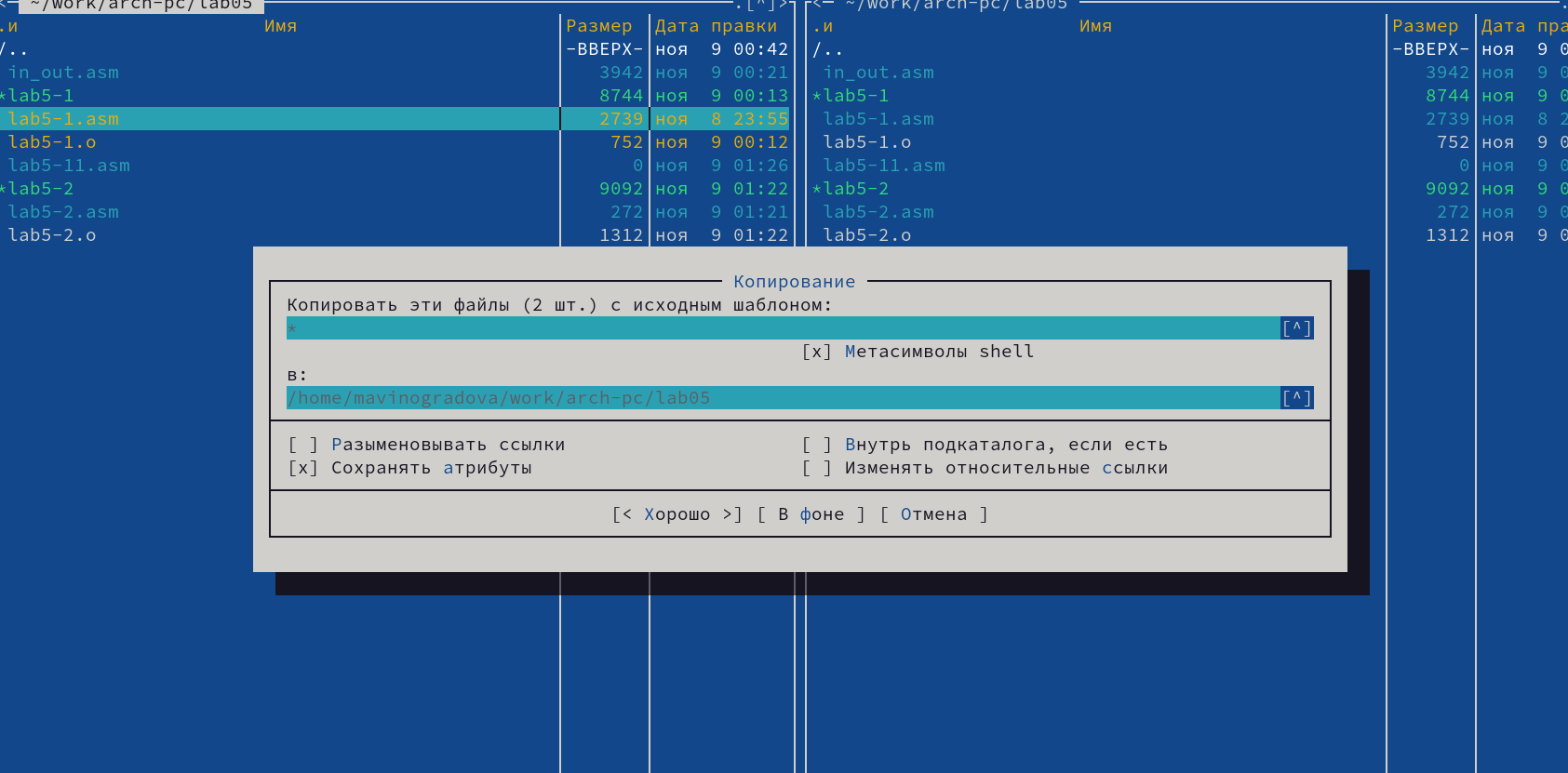


Рис. 16: Создаем копию файла lab5-1.asm

Редактируем файл, чтобы введеный текст с клавиатуры выводился в консоль (рис. 17).

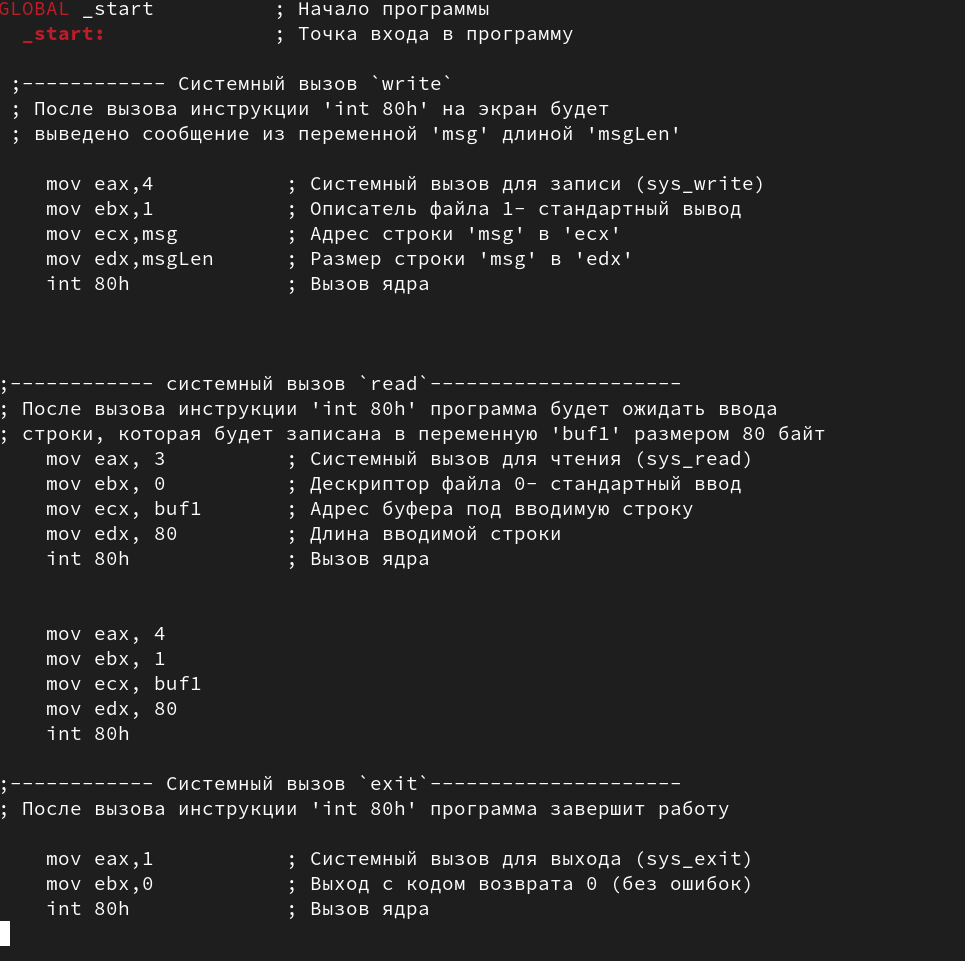


Рис. 17: Редактируем файл

Транслируем файл и запускаем программу (рис. 18).

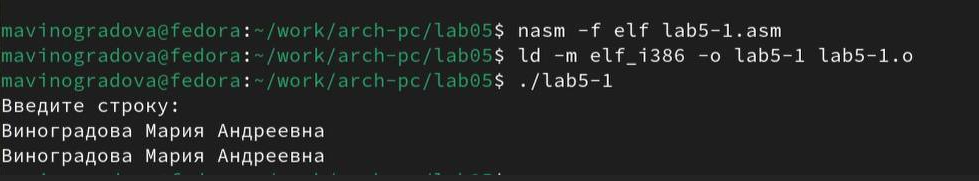


Рис. 18: Проверяем правильность написания программы

Создаем копию файла lab5-2.asm и называем его так же (рис. 19).

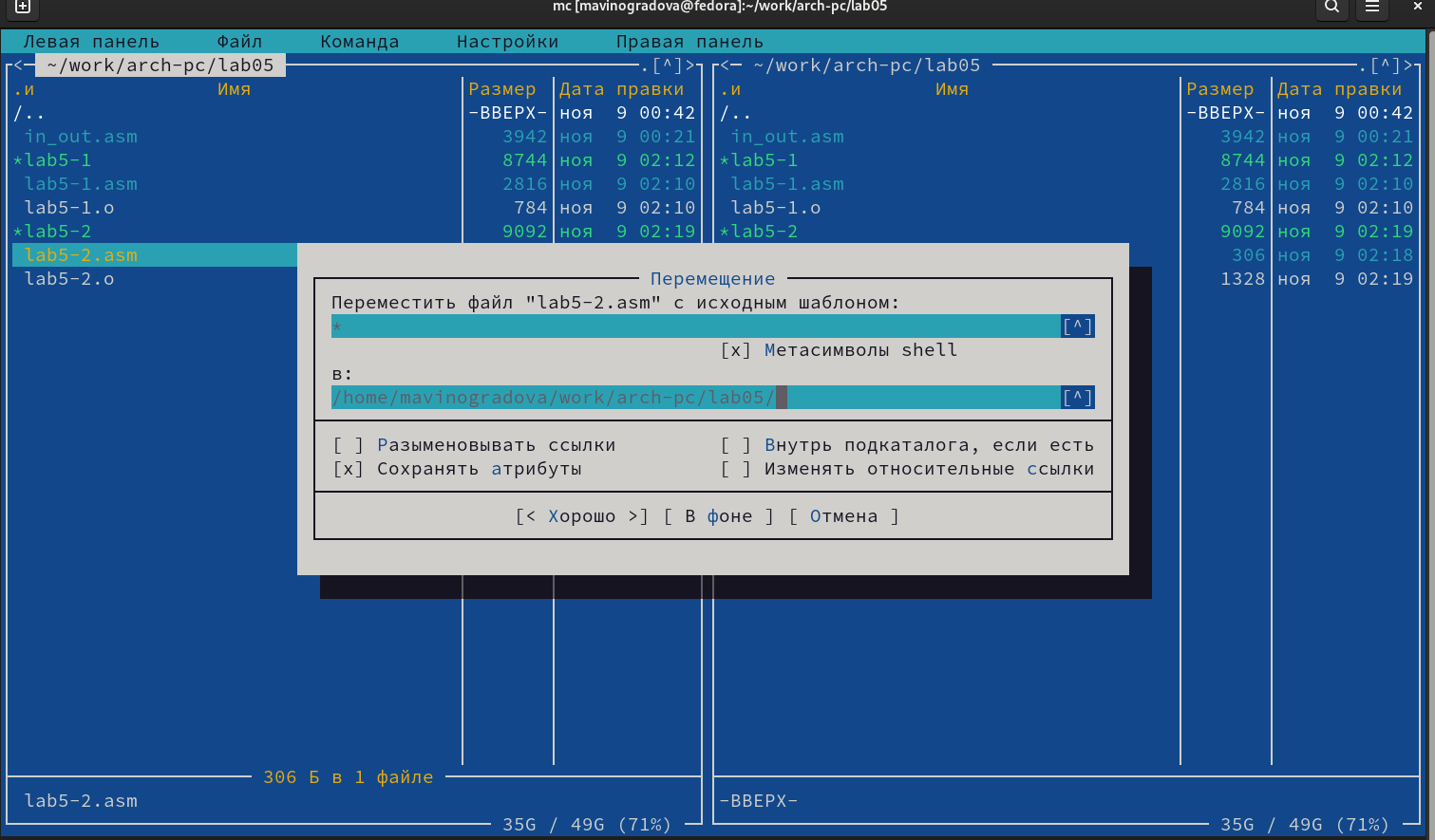


Рис. 19: Создаем копию файла lab5-2.asm

Редактируем файл, чтобы введеный текст с клавиатуры выводился в консоль (рис. 20).

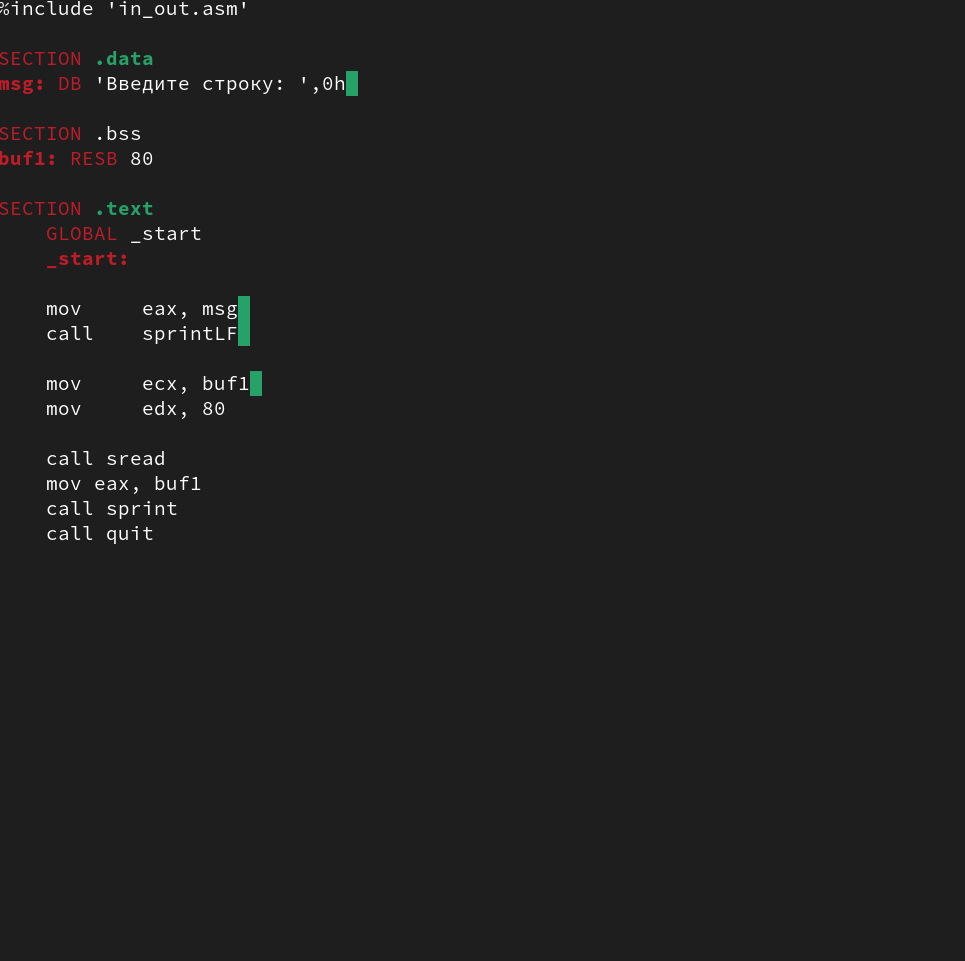


Рис. 20: Редактируем файл

Транслируем файл и запускаем программу (рис. 21).

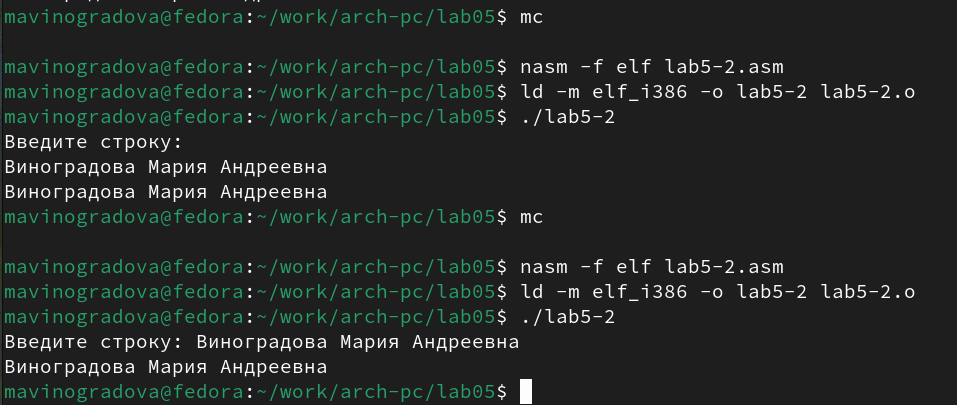


Рис. 21: Проверяем правильность написания программы

# 5 Выводы

Мы приобрели навыки работы с Midnight Commander и освоили инструкцию mov.