Отчёт по лабораторной работе №5

Основы работы с Midnight Commander (mc). Структура программы на языке ассемблера NASM. Системные вызовы в ОС GNU Linux

Грачева Мария Валерьевна

Содержание

# 1 Цель работы

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

# 2 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).

Простейший диалог с пользователем требует наличия двух функций — вывода текста на экран и ввода текста с клавиатуры. Простейший способ вывести строку на экран — использо- вать системный вызов write. Этот системный вызов имеет номер 4, поэтому перед вызовом инструкции int необходимо поместить значение 4 в регистр eax. Первым аргументом write, помещаемым в регистр ebx, задаётся дескриптор файла. Для вывода на экран в качестве дескриптора файла нужно указать 1 (это означает «стандартный вывод», т. е. вывод на экран). Вторым аргументом задаётся адрес выводимой строки (помещаем его в регистр ecx, напри- мер, инструкцией mov ecx, msg). Строка может иметь любую длину. Последним аргументом (т.е. в регистре edx) должна задаваться максимальная длина выводимой строки. Для ввода строки с клавиатуры можно использовать аналогичный системный вызов read. Его аргументы – такие же, как у вызова write, только для «чтения» с клавиатуры используется файловый дескриптор 0 (стандартный ввод). Системный вызов exit является обязательным в конце любой программы на языке ассем- блер. Для обозначения конца программы перед вызовом инструкции int 80h необходимо поместить в регистр еах значение 1, а в регистр ebx код завершения 0.

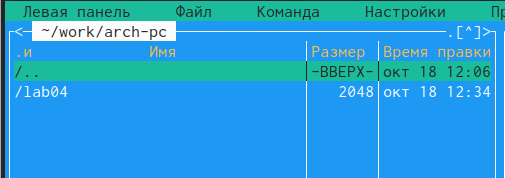
# 3 Выполнение лабораторной работы

Открываю Midnight Commander (рис. ??).

Команда mc

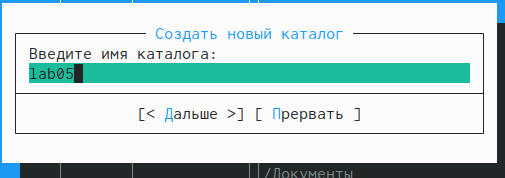
Команда mc

Перехожу в каталог arch-pc (рис. ??).



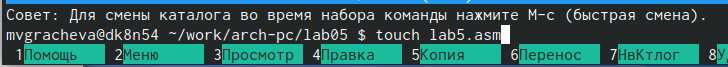
Каталог arch-pc

Создаю папку lab05 (рис. ??).



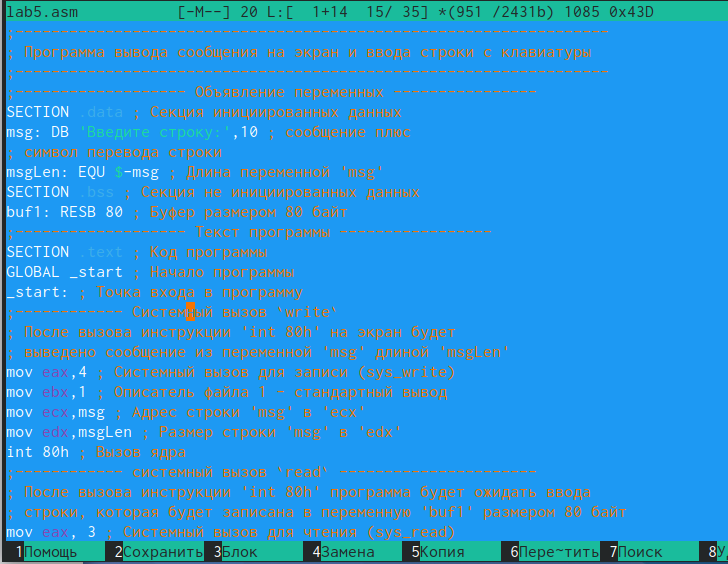
Создание папки lab05

Создаю файл lab5.asm (рис. ??).



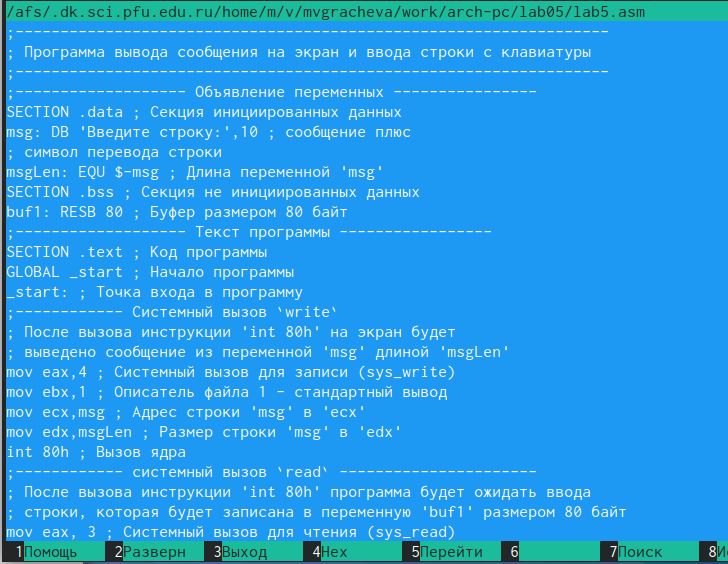
Создание файла lab5.asm

Открываю этот файл и ввожу текст программы (рис. ??).



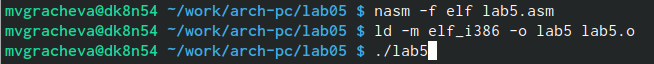
Ввод текста программы

С помощью функциональной клавиши f3 проверяю текст (рис. ??).



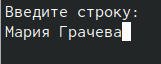
Проверка текста

Транслирую текст программы lab5.asm в объектный файл. Выполняю компо- новку объектного файла и запускаю получившийся исполняемый файл (рис. ??).



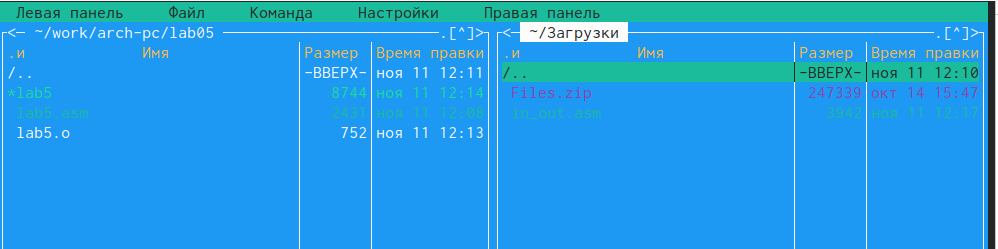
Транслирование, компновка, запуск lab5.asm

Ввожу своё ФИО (рис. ??).



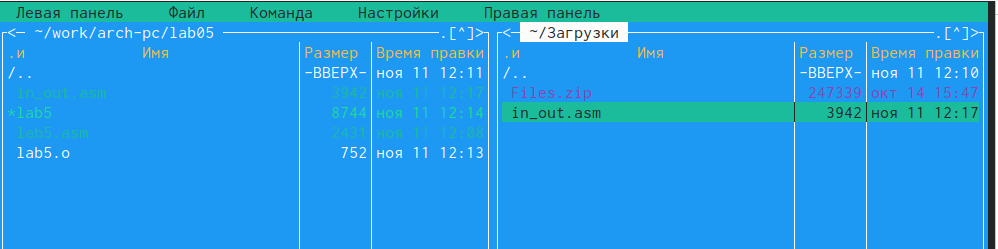
Ввод имени

Открываю два разных каталога одновременно на правой и левой панелях (рис. ??).

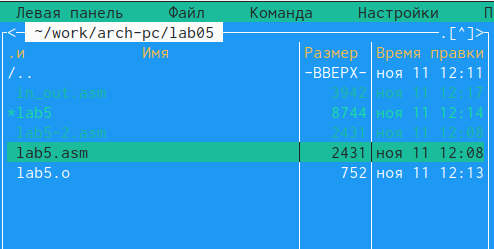


Работа с панелями

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

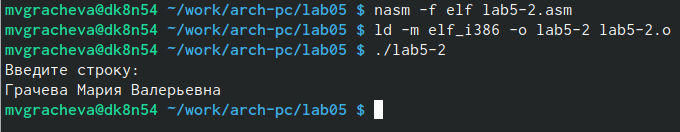


Создаю копию файла lab5-1.asm с именем lab5-2.asm (рис. ??).



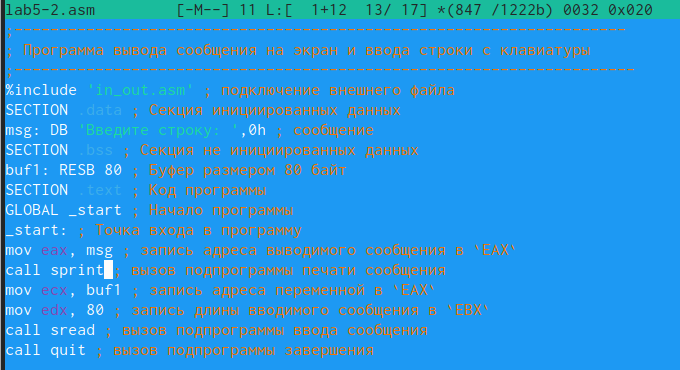
Копирование файла lab5-1.asm с именем lab5-2.asm

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. ??).



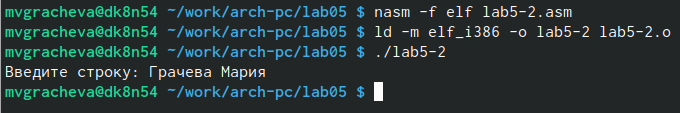
Изменение текста, транслирование, компновка, запуск lab5-2.asm

Заменяю подпрограмму sprintLF на sprint (рис. ??).



Замена подпрограммы

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

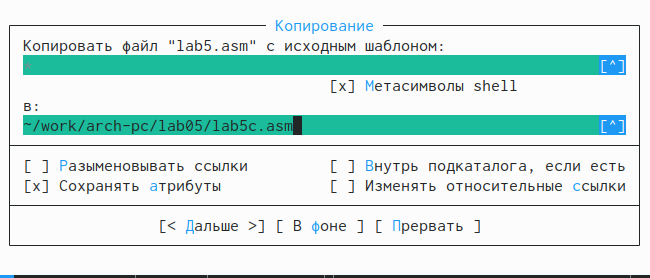


Транслирование, компновка, запуск lab5-2.asm. 2

Разница заключается в переносе строки ввода

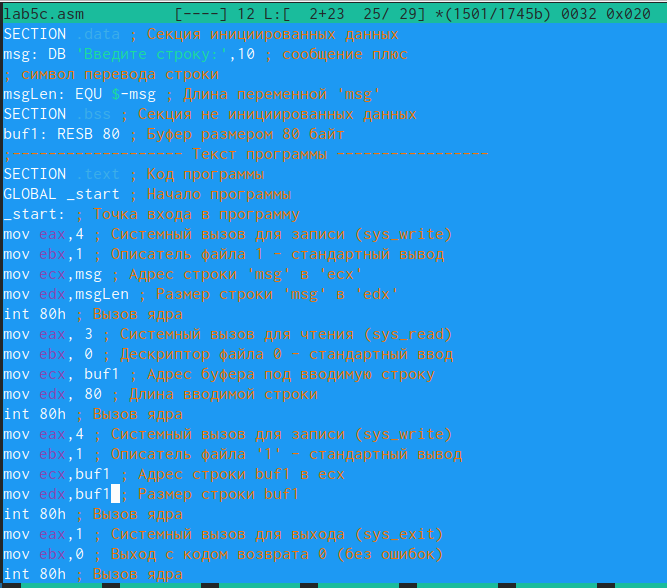
# 4 Самостоятельная работа

1. Создаю копию файла lab5.asm (рис. ??).



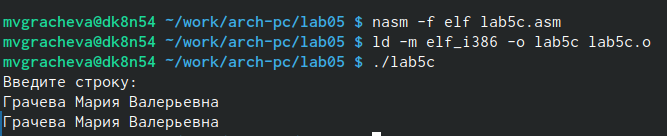
Копирование файла lab5.asm

Вношу изменения в программу (рис. ??).



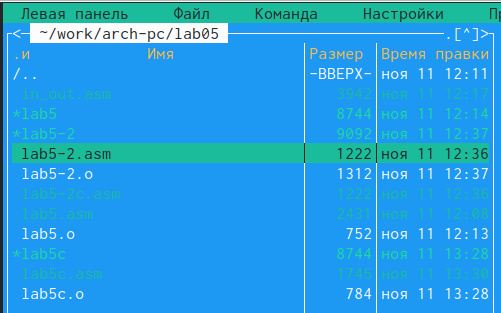
Изменение текста. 2

Получаю исполняемый файл и проверяю его работу (рис. ??).



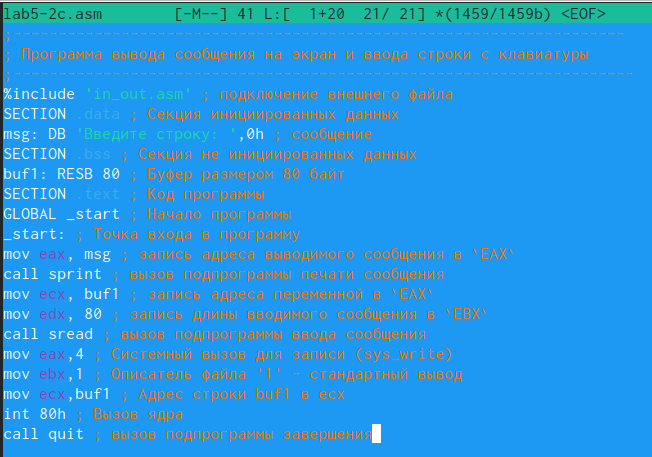
Транслирование, компновка, запуск lab5с.asm

Создаю копию файла lab5-2.asm (рис. ??).



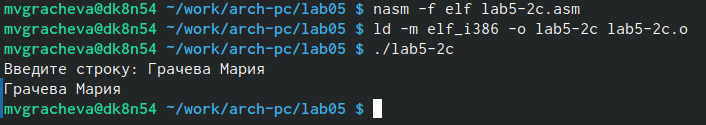
Копирование файла lab5-2.asm

Вношу изменения в программу (рис. ??).



Изменение текста. 3

Получаю исполняемый файл и проверяю его работу (рис. ??).



Транслирование, компновка, запуск lab5-2с.asm

# 5 Выводы

Я приобрела практические навыки работы в Midnight Commander. Освоила инструкции языка ассемблера mov и int.

# Список литературы