Отчёт по лабораторной работе №5

Основы работы с Midnight commander. Структура программы на языке NASM. Системные вызовы

Ведьмина Александра Сергеевна

Содержание

# 1 Цель работы

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

# 2 Задание

1. Изучить основы работы с mc.
2. Познакомиться со структурой программы вывода сообщений на экран и ввода со строки на языке NASM.
3. Подключить файл in\_out.asm.
4. Выполнить задания для самостоятельной работы.

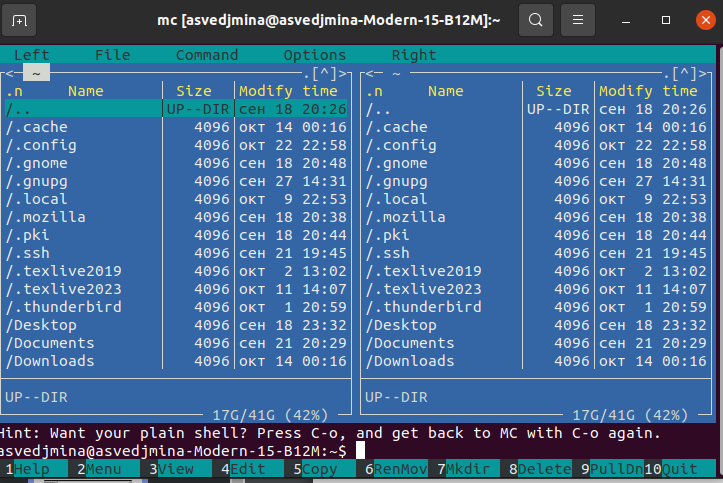
# 3 Теоретическое введение

Midnight Commander — это файловый менеджер, с помощью которого можно просматривать структуру каталогов и выполнять основные операции по управлению файловой системой. Чтобы начать использовать Midnight Commander, достаточно ввести в командной строке mc. Программа на языке ассемблера NASM обычно состоит из трёх секций: секция кода, секция инициированных данных и секция неинициализированных данных. 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. Директивы применяются для объявления простых переменных и для объявления массивов. Для определения строк принято использовать директиву DB. Для объявления неинициированных данных в секции .bss используются директивы resb, resw, resd и другие. В качестве операндов могут выступать регистры, ячейки памяти и значения.

Простейший способ вывести строку на экран — использовать 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 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.

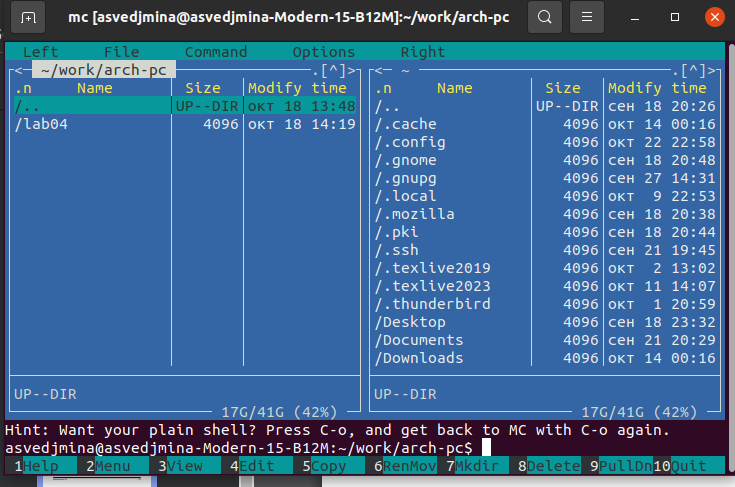
# 4 Выполнение лабораторной работы

Открываю Midnight Commander.



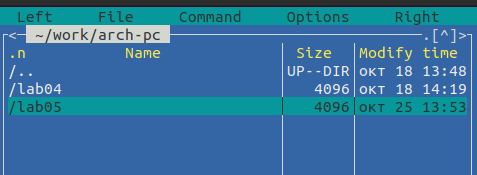
Открытие mc

Перехожу в каталог, созданный при выполнении лабораторной №4.



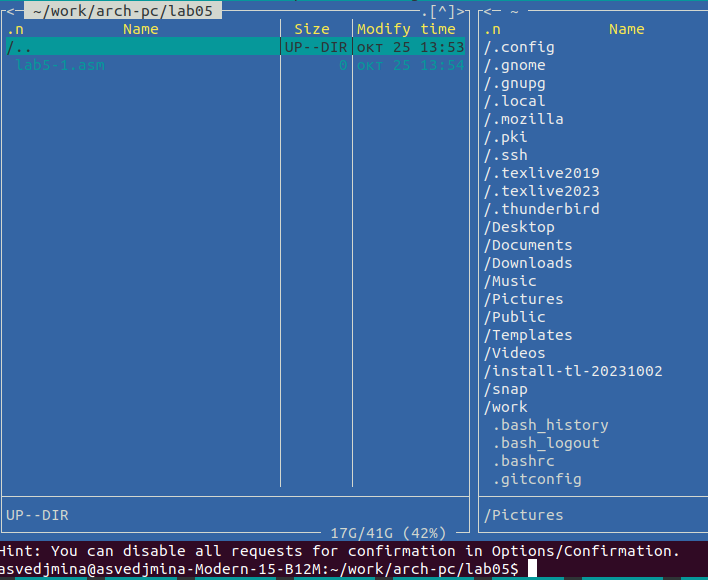
Переход в каталог ~/work/arch-pc

Создаю папку lab05 и перехожу в созданный каталог.



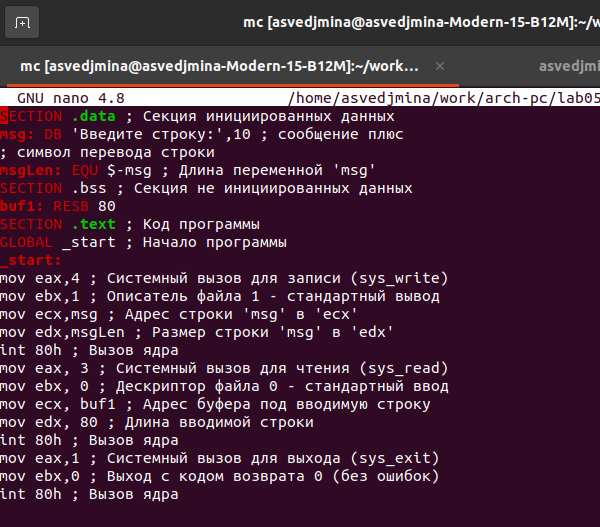
Создание папки lab05

Создаю файл lab5-1.asm.



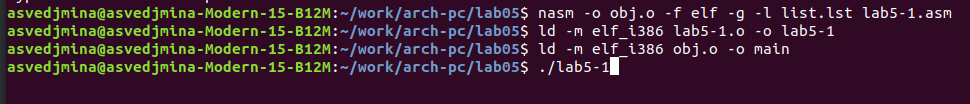
Создание файла lab5-1.asm

Открываю этот файл в редакторе nano и ввожу предагаемый текст программы, после чего закрываю файл.

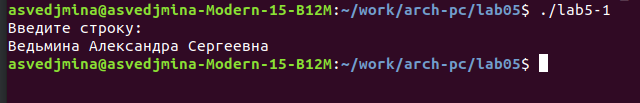


Открытие mc

Транслирую lab5-1.asm в объектный файл, выполняю его компановку и запускаю исполняемый файл. На запрос ввожу своё ФИО (Ведьмина Александра Сергеевна).

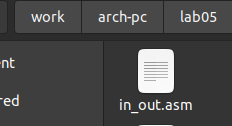


Создание исполняемого файла lab5-1



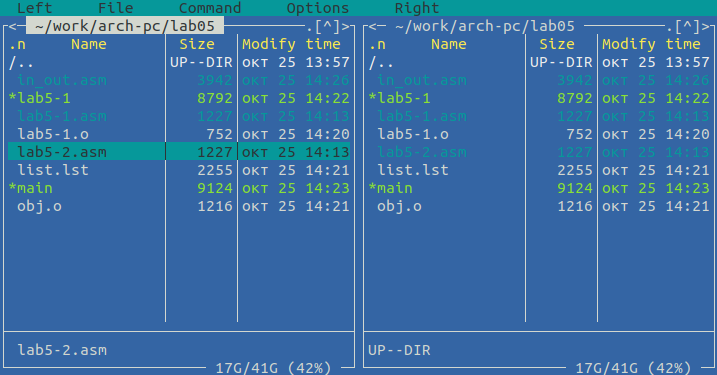
Запуск lab5-1

Далее скачиваю файл in\_out.asm с ТУИСа и помещаю его в ~/work/arch-pc/lab05.



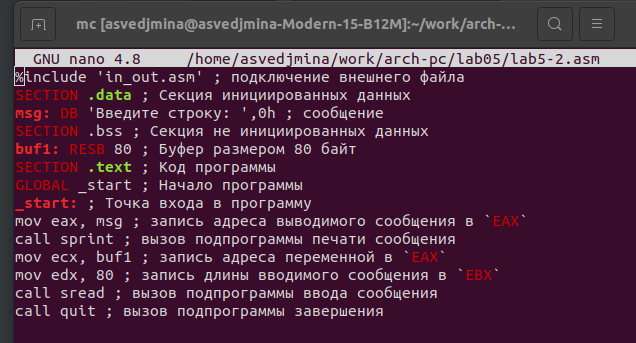
Запуск lab5-1

Делаю копию файла lab5-1.asm и называю её lab5-2.asm.



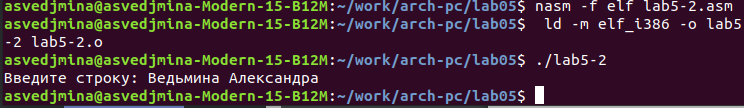
Создание копии lab5-1

Используя подпрограммы из in\_out.asm, вношу изменения в lab5-2.asm.



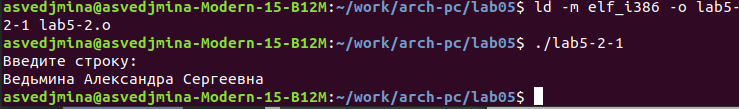
Изменение программы в lab5-2

Создаю исполняемый файл и запускаю его.



Запуск lab5-2

В файле lab5-2.asm заменяю sprintLF на sprint, создаю исполняемый файл lab5-2-1 и запускаю его, после чего переименовываю его в lab5-2.

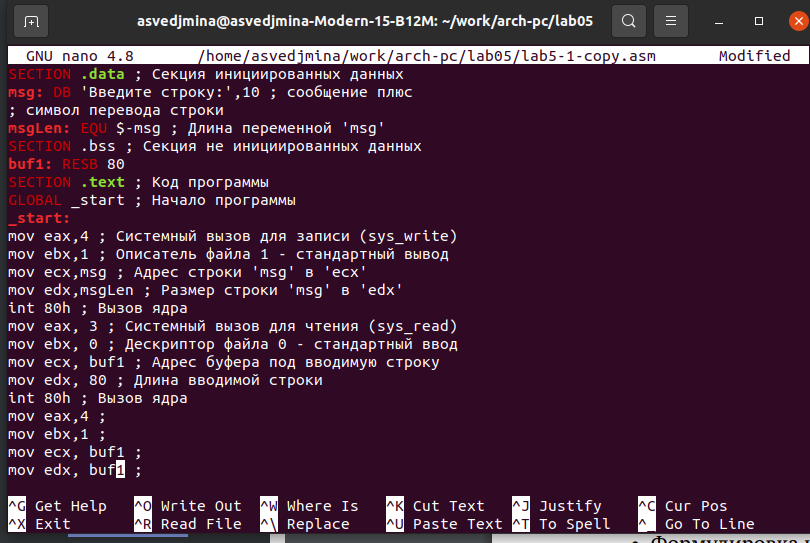


Запуск lab5-2-1

Разница между программами со sprintLF и sprint состоит в том, что просит ввести данные на той же строке, на которой выводится их запрос, в то время как вторая предлагает сделать это на новой строке.

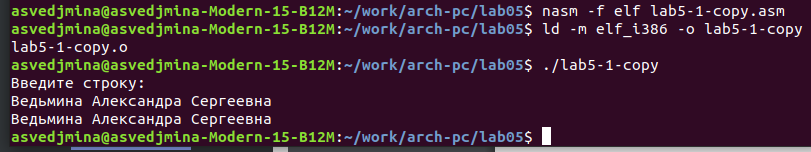
# 5 Выполнение заданий для самостоятельной работы

Создаю копию файла lab5-1.asm с именем lab5-1-copy.asm и вношу изменения в программу так, чтобы она выводила введённую строку на экран.



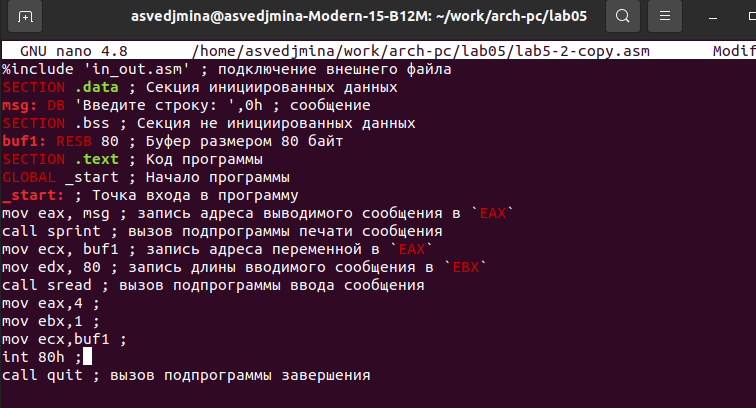
Внесение изменений в файл lab5-1-copy.asm

Получаю исполняемый файл и запускаю его, вводя на запрос своё ФИО.



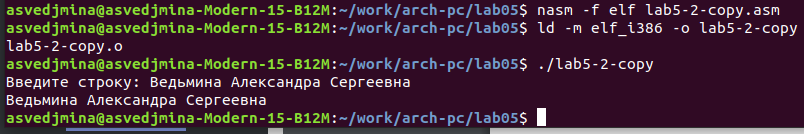
Запуск lab5-1-copy

Создаю копию файла lab-2.asm с именем lab5-2-copy.asm, изменяю программу так, чтобы она тоже выводила на экран введённую строку.



Внесение изменений в файл lab5-2-copy.asm

После этого создаю исполняемый файл и запускаю его.



Запуск lab5-2-copy

# 6 Выводы

В ходе этой лабораторной работы я научилась в Midnight Commander и освоила инструкции языка ассемблера mov и int.