Отчёт по лабораторной работе 5

Архитектура компьютеров и операционные системы

Горелашвили Лия Михайловна НКАбд-02-23

Содержание

# 1 Цель работы

Целью работы является приобретение практических навыков работы в Midnight Commander. Освоение инструкций языка ассемблера mov и int.

# 2 Задание

1. Изучение возможностей Midnight Commander
2. Изучение файла in\_out.asm
3. Выполнение заданий, рассмотрение примеров
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 неинициализированных данных (тех, под которые во время компиляции только отводится память, а значение присваивается в ходе выполнения программы) (SECTION .bss).

# 4 Выполнение лабораторной работы

## 4.1 Знакомство с Midnight Commander

Открыла Midnight Commander и с помощью клавиш со стрелками и Enter перешла в каталог ~/work/arch-pc. Затем нажала F7 и создала каталог с названием lab05.

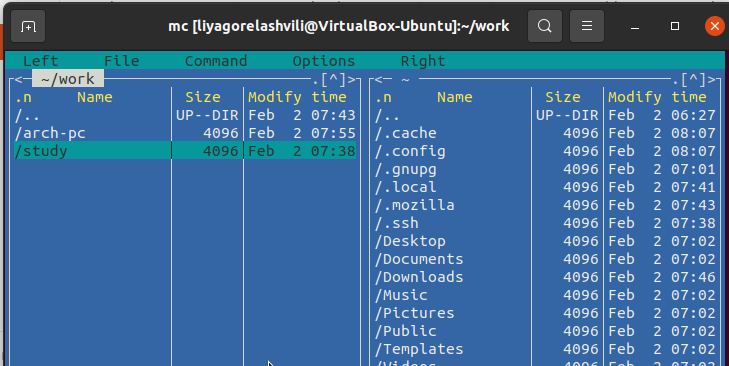


Figure 1: окно Midnight Commander

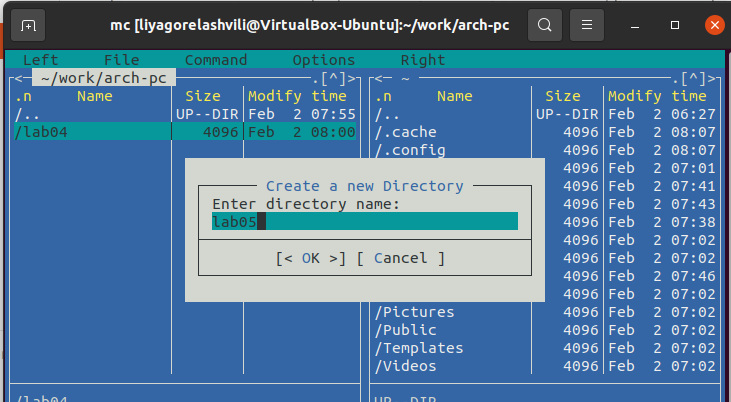


Figure 2: Создание каталога

Используя команду touch, создала файл с именем lab05-1.asm.

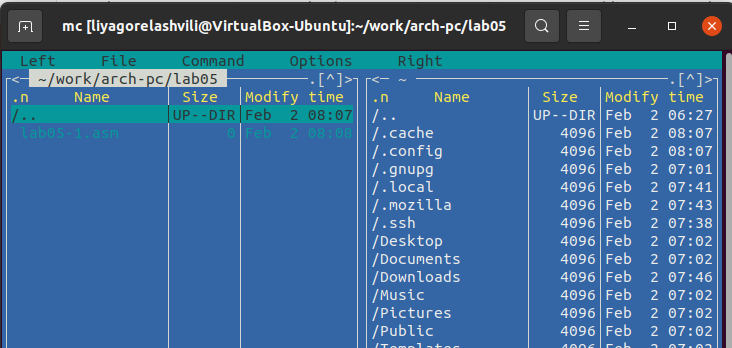


Figure 3: Создала файл lab05-1.asm

Затем открыла файл для редактирования, нажав клавишу F4, и выбрала редактор mceditor. Написала код программы, соответствующий заданию.

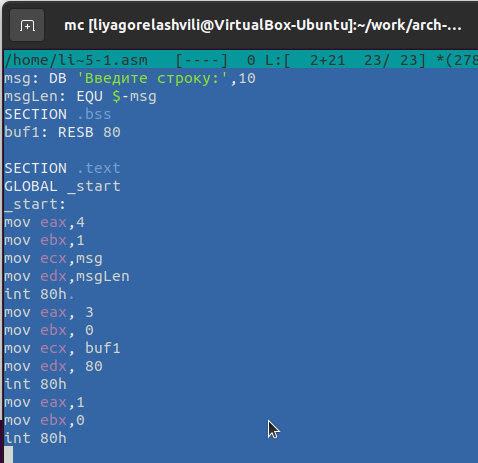


Figure 4: Редактирование файла lab05-1.asm

Далее открыла файл для просмотра, нажав клавишу F3, и убедилась, что он содержит написанный код.

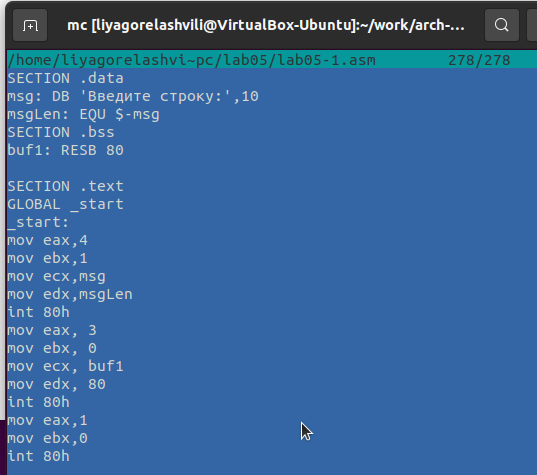


Figure 5: Проверка кода lab05-1.asm

С помощью трансляции файла программы в объектный файл, выполнения компоновки объектного файла и получения исполняемого файла, проверила работу программы.

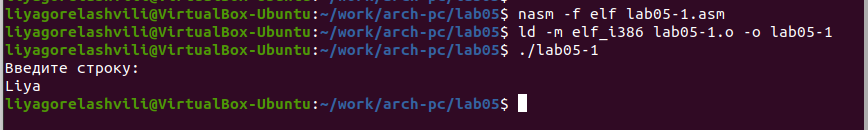


Figure 6: Тестирование программы lab05-1.asm

## 4.2 Подключение внешнего файла in\_out.asm

Скачала файл “in\_out.asm” и разместила его в рабочем каталоге.

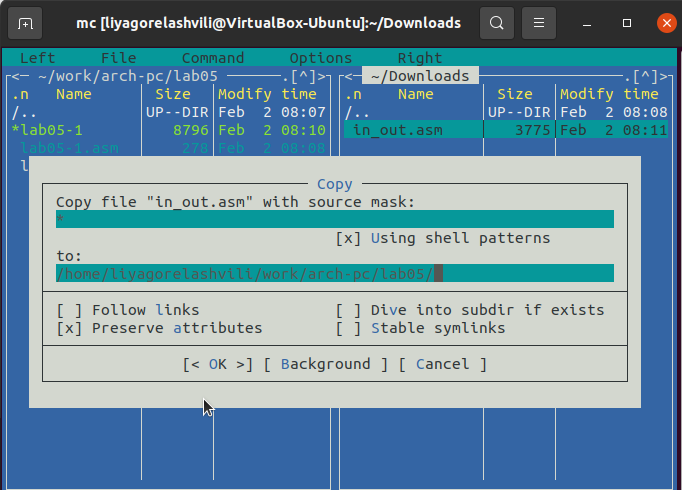


Figure 7: Копирование файла in\_out.asm

С помощью клавиши F5 скопировала содержимое файла lab05-1.asm в файл lab05-2.asm.

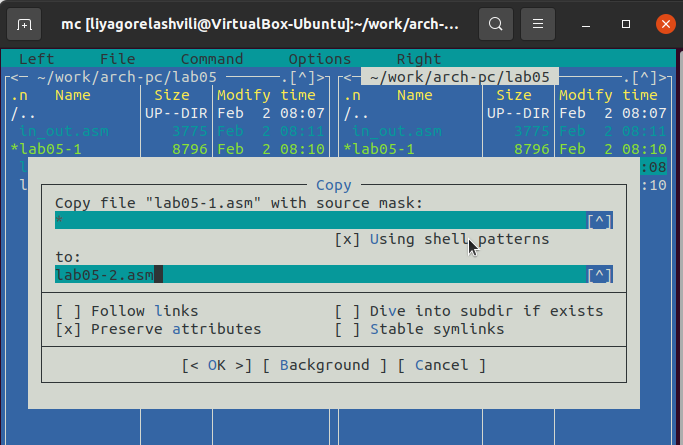


Figure 8: Копирование файла lab05-1.asm

Затем написала код программы lab05-2.asm, используя подпрограммы из внешнего файла in\_out.asm.

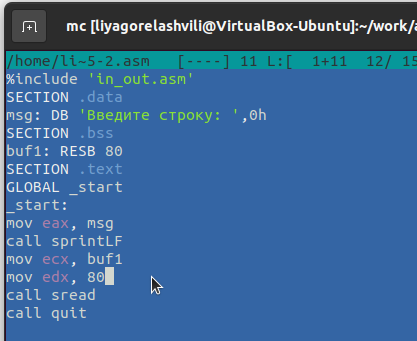


Figure 9: Редактирование файла lab05-2.asm

Скомпилировала программу и проверила ее запуск.

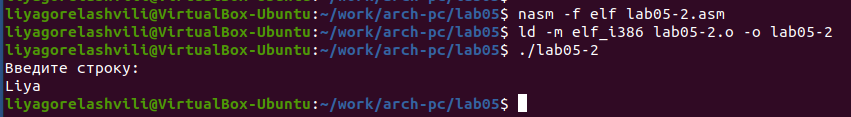


Figure 10: Тестирование программы lab05-2.asm

В файле lab05-2.asm заменила вызов подпрограммы sprintLF на sprint. Пересобрала исполняемый файл. Теперь после вывода строки символ перехода на новую строку отсутствует.

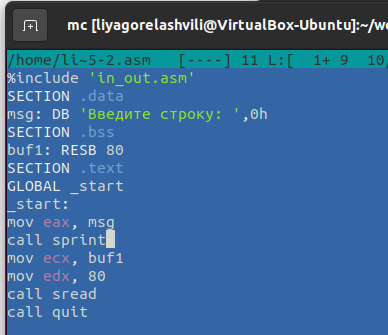


Figure 11: Редактирование файла lab05-2.asm

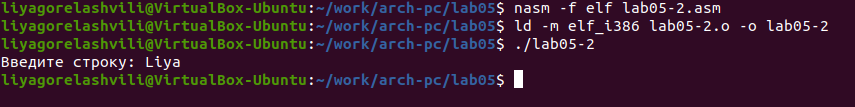


Figure 12: Тестирование программы lab05-2.asm

## 4.3 Выполнение заданий для самостоятельной работы

Скопировала программу lab05-1.asm и внесла изменения в код, чтобы программа работала по следующему алгоритму: она выводит приглашение вида “Введите строку:”, считывает строку с клавиатуры и выводит введенную строку на экран.

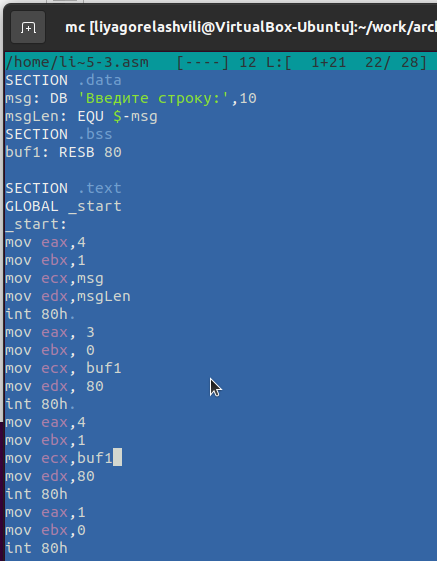


Figure 13: Редактирование файла lab05-3.asm

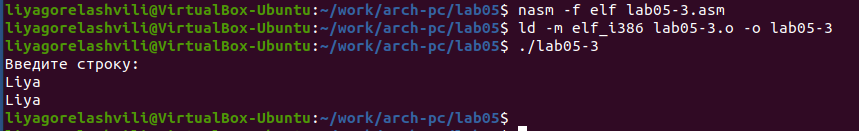


Figure 14: Тестирование программы lab05-3.asm

Аналогично скопировала программу lab05-2.asm и изменила код, но теперь использовала подпрограммы из файла in\_out.asm.

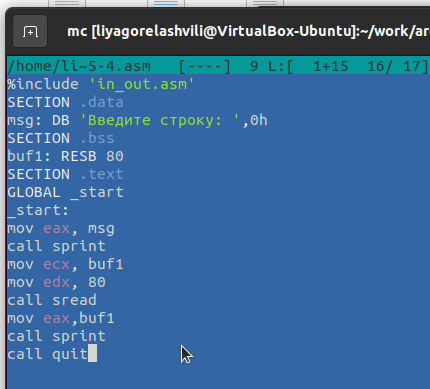


Figure 15: Редактирование файла lab05-4.asm

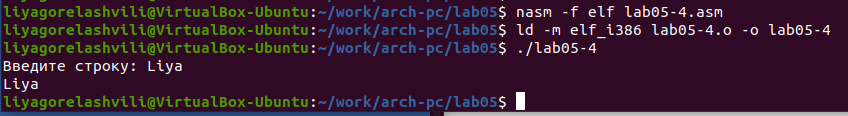


Figure 16: Тестирование программы lab05-4.asm

# 5 Выводы

Научились писать базовые ассемблерные программы. Освоили ассемблерные инструкции mov и int.