Лабораторная работа №6

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

Татьяна Алексеевна Коннова, НПИбд-01-22

Содержание

[1 Цель работы 1](#__RefHeading___Toc324_1849107431)

[2 Задание 1](#__RefHeading___Toc326_1849107431)

[2.1 Выполнение лабораторной работы : Программа Hello world! 1](#__RefHeading___Toc328_1849107431)

[3 Выводы 9](#__RefHeading___Toc330_1849107431)

# 1 Цель работы

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

# 2 Задание

Освоение инструкций языка ассемблера mov и int.

## 2.1 Выполнение лабораторной работы : Программа Hello world!

Перейдем в Midnight Commander.

(рис. 1)

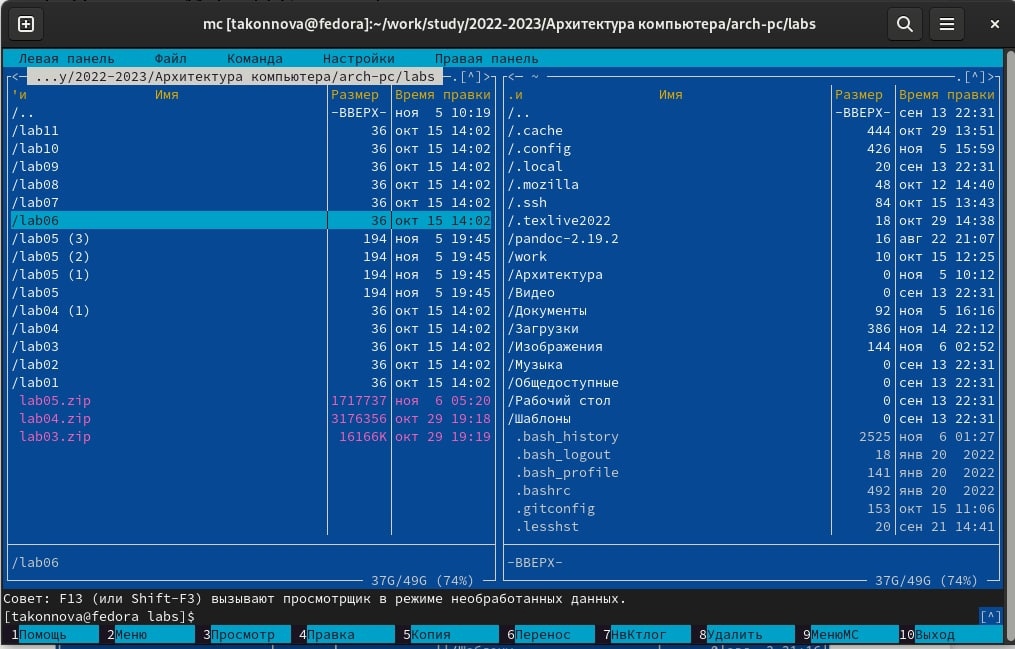


Рис. 1: Переход

Создадим с помощью команды touch файл lab6-2.asm (рис. 2)

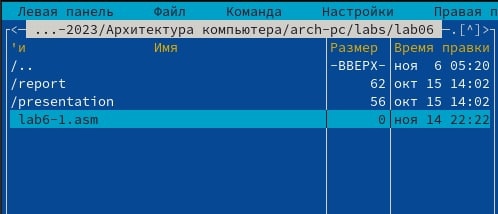


Рис. 2: lab6-2.asm

Оттранслируем текст программы lab6-1.asm в объектный файл. Выполним компоновку объектного файла и запустиим получившийся исполняемый файл. Программа выводит строку ‘Введите строку:’ и ожидает ввода с клавиатуры. На запрос введем наше имя. (рис. 3)

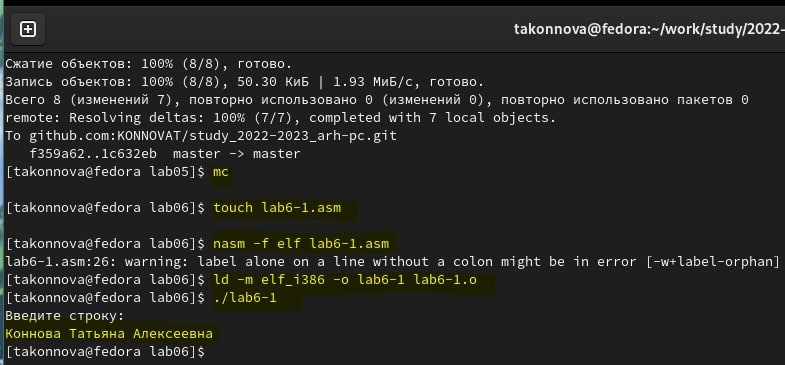


Рис. 3: Текст с последующим выводом необходимого

Скачаем файл in\_out.asm со страницы курса в ТУИС. Подключаемый файл in\_out.asm должен лежать в том же каталоге, что и файл с программой, в которой он используется. В одной из панелей mc откроем каталог с файлом lab6-1.asm. В другой панели каталог со скаченным файлом in\_out.asm (для перемещения между панелями используем Tab ). Скопируем файл in\_out.asm в каталог с файлом lab6-1.asm с помощью функциональной клавиши

(рис. 4)

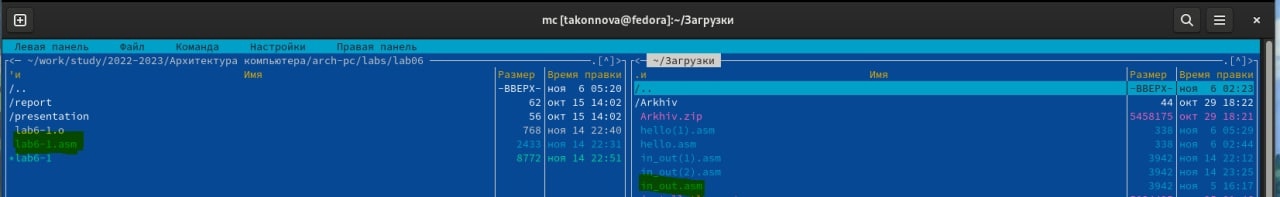


Рис. 4: Действия с in\_out.asm

С помощью функциональной клавиши F6 создаем копию файла lab6- 1.asm с именем lab6-2.asm. Выделяем файл lab6-1.asm, нажимаем клавишу F6 , введем имя файла lab6-2.asm и нажмем клавишу Enter

(рис. 5)

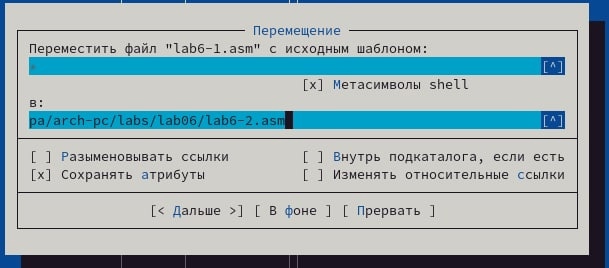


Рис. 5: Копирование

Исправим текст программы в файле lab6-2.asm с использование под- программ из внешнего файла in\_out.asm (используем подпрограммы sprintLF, sread и quit) в соответствии с листингом 6.2. Создаем испол- няемый файл и проверяем его работу (рис. 6)

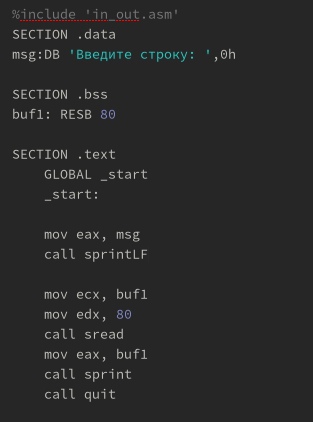


Рис. 6: Проверка

Проеверяем результат работы программы lab6-2.asm (рис. 7)

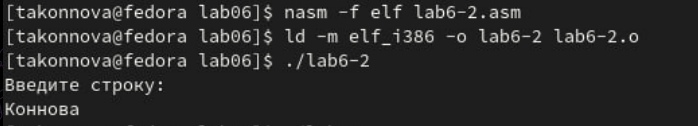


Рис. 7: Результат

При включении LF в Sprint вводимые данные отображаются для ввода на следующей строчке. ## Задание для самостоятельной работы

1. Создаем копию файла lab6-1.asm. Внесем изменения в программу (без использования внешнего файла in\_out.asm), так чтобы она работала по следующему алгоритму:

вывести приглашение типа “Введите строку:”;

ввести строку с клавиатуры;

вывести введённую строку на экран.

Пусть копией lab6-1.asm будет являться lab6-3.asm (рис. 8)

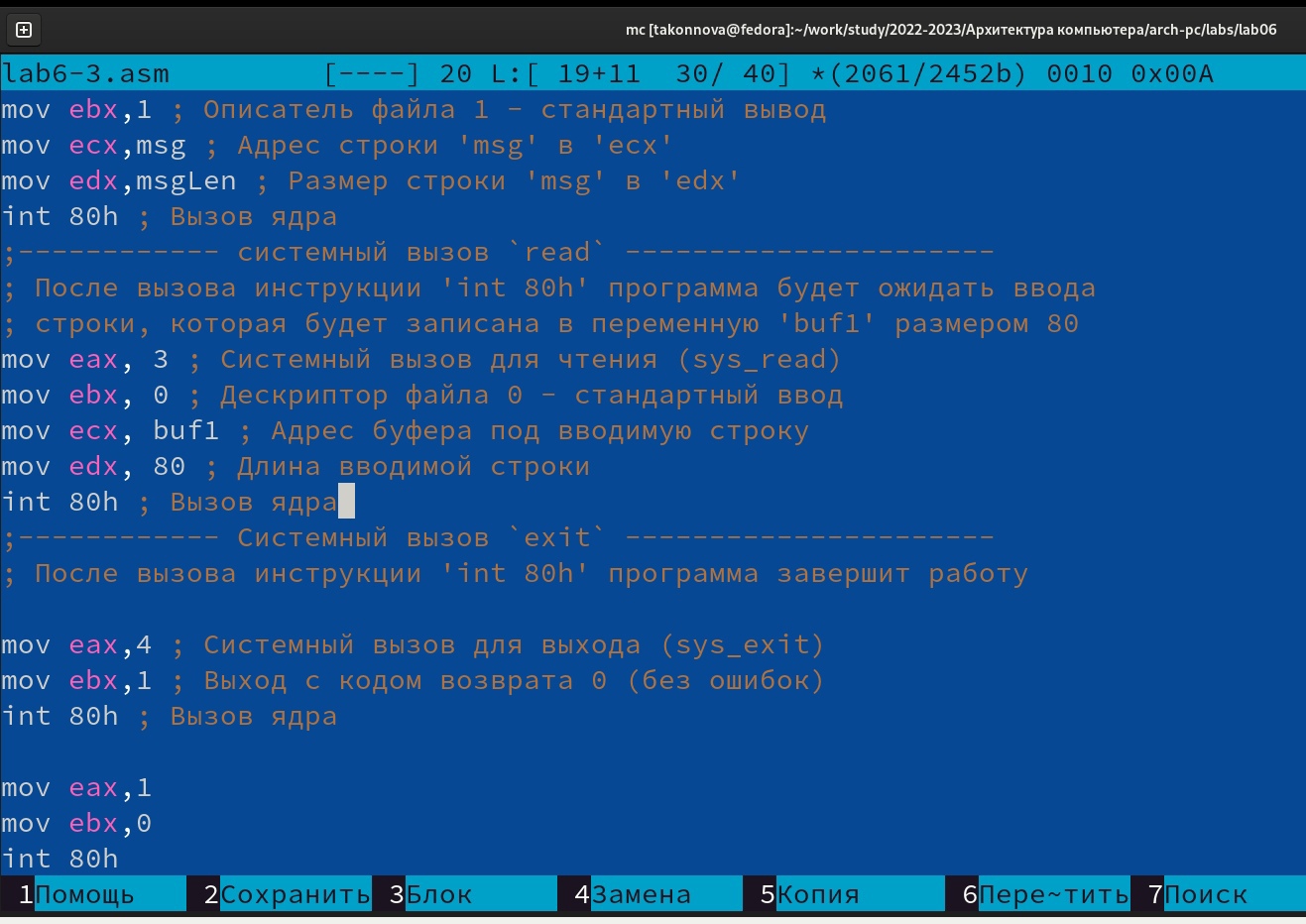


Рис. 8: Вводимый и измененный текст

1. Получаем исполняемый файл и проверяем его работу. На приглашение ввести строку введем свою фамилию. (рис. 9)

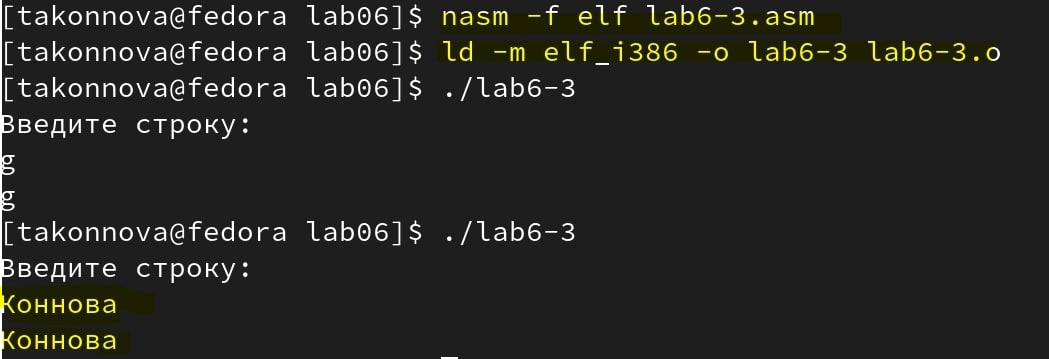


Рис. 9: Проверка

1. Создаем копию файла lab6-2.asm. Исправляем текст программы с исполь- зование подпрограмм из внешнего файла in\_out.asm, так чтобы она ра- ботала по следующему алгоритму:

вывести приглашение типа “Введите строку:”;

ввести строку с клавиатуры;

вывести введённую строку на экран.

Подключаемый файл in\_out.asm должен лежать в том же каталоге, что и файл с программой, в которой он используется. Пусть имя данного файла будет lab6-4.asm (рис. 10)

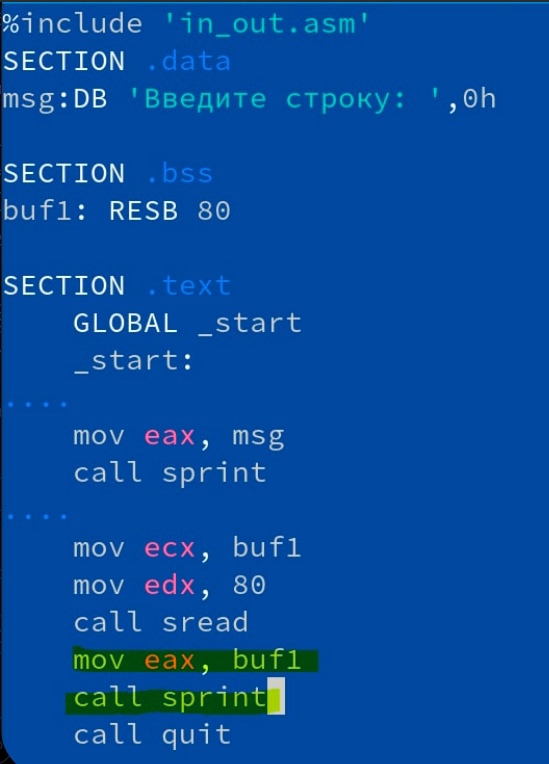


Рис. 10: Изменение файла

1. Создаем исполняемый файл и проверяем его работу

(рис. 11)

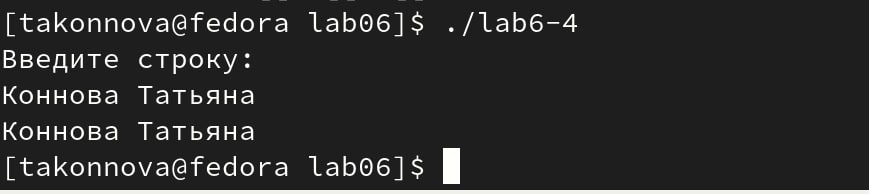


Рис. 11: Проверка

# 3 Выводы

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