Лабораторная работа №5

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

Коровкин Никита Михайлович

Содержание

# 1 Цель работы

Ознакомиться с программой Midnight commander и освоить написание программ на языке ассемблера с помощью инструкций mov и int

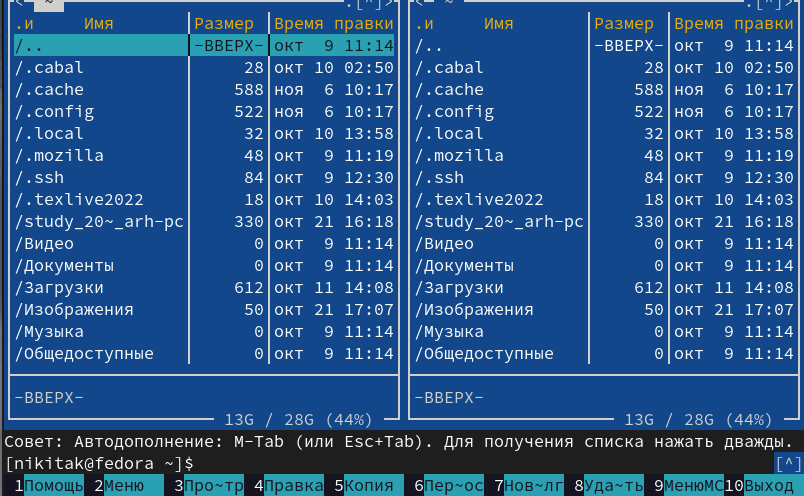
# 2 Выполнение лабораторной работы

Для начала выполнения лабораторной работы нам нужно открыть Midnight commander с помощью команды mc (Рис.1):

Запуск Midnight commander при помощи mc

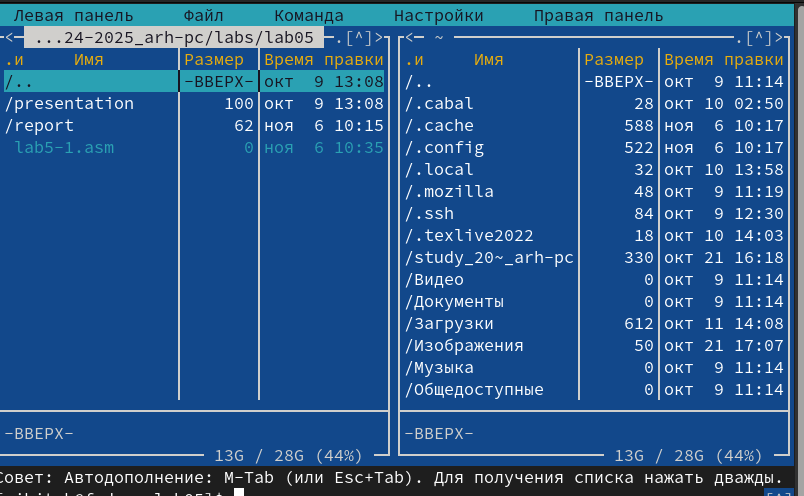
Запуск Midnight commander при помощи mc

Перед нами появится такой интерфейс (Рис.2)



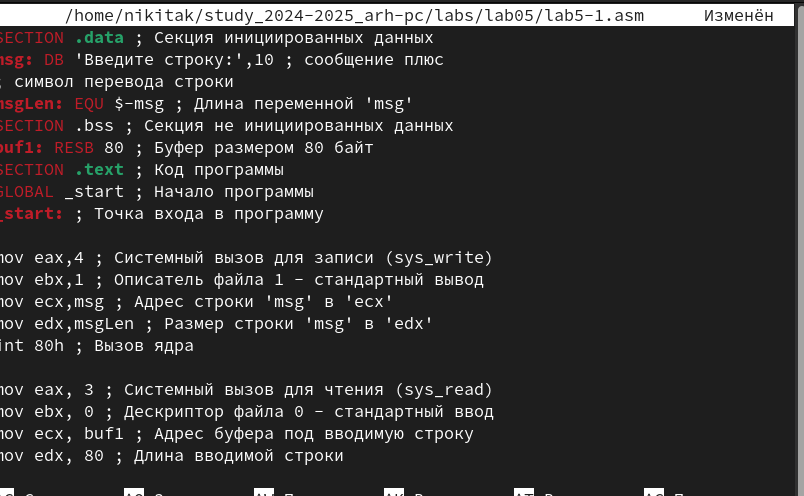
Midnight commander

Создадим файл lab5-1.asm в нужной директории. (Рис.3)



файл lab5-1.asm

Откроем созданный файл и отредактируем его, вставив туда нужный код. (Рис.4)



Вставляем нужный код в файл

Сохранив его, скомпилируем файл. (Рис.5)

Компиляция кода

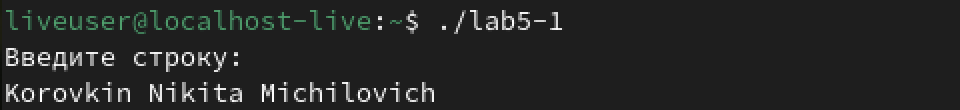
Компиляция кода

Теперь соберем файл. (Рис.6)

Сборка файла

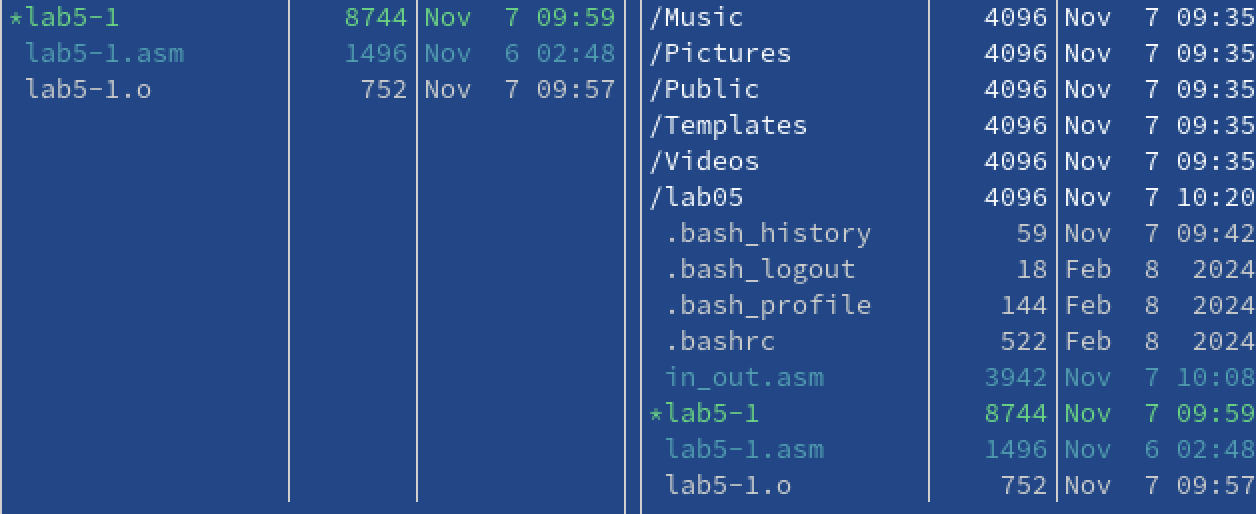
Сборка файла

После этого запустим получившийся исполняемый файл. (Рис. 7):



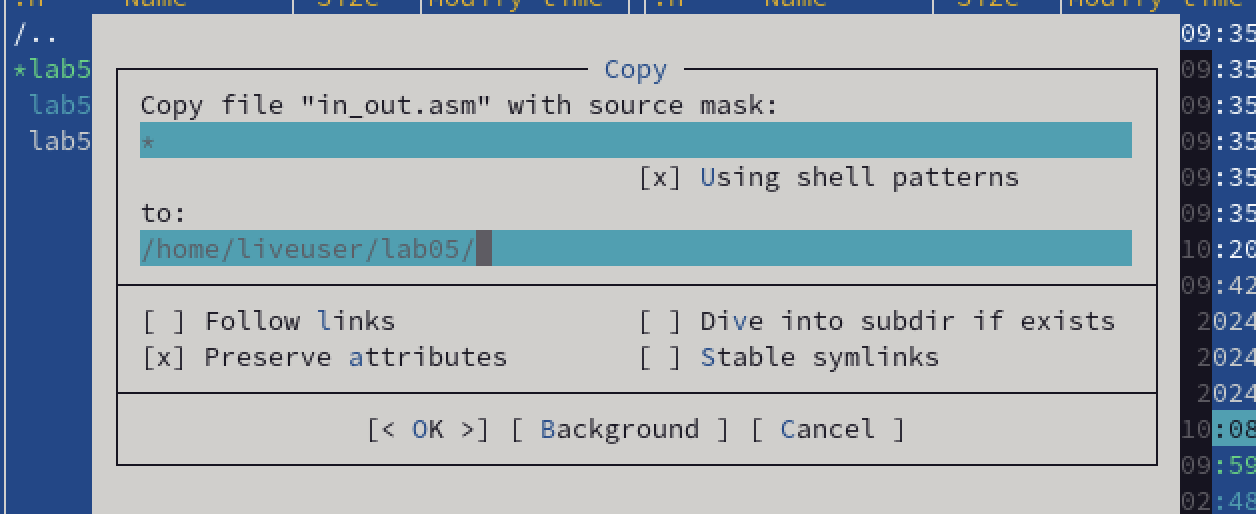
Запуск исполняемого файла

Запишем ФИО в строку и нажмем Enter. Ничего не произойдет. Тогда скачаем файл in\_out.asm и откроем Midnight Commander. (Рис.8)



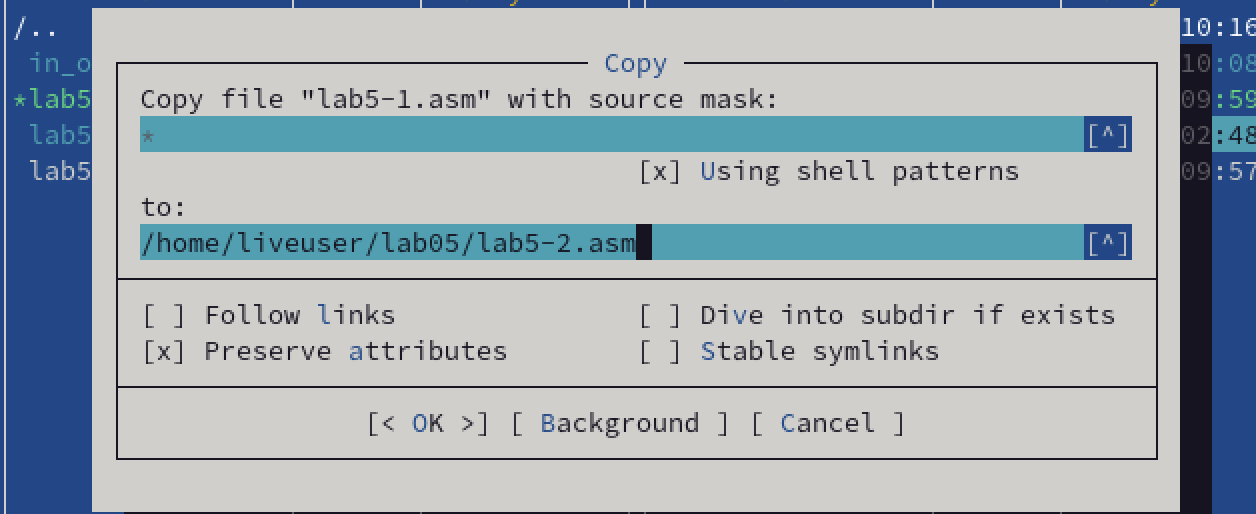
Файл in\_out.asm

Скопируем файл в нужную папку, с которой мы работаем. (Рис.9)



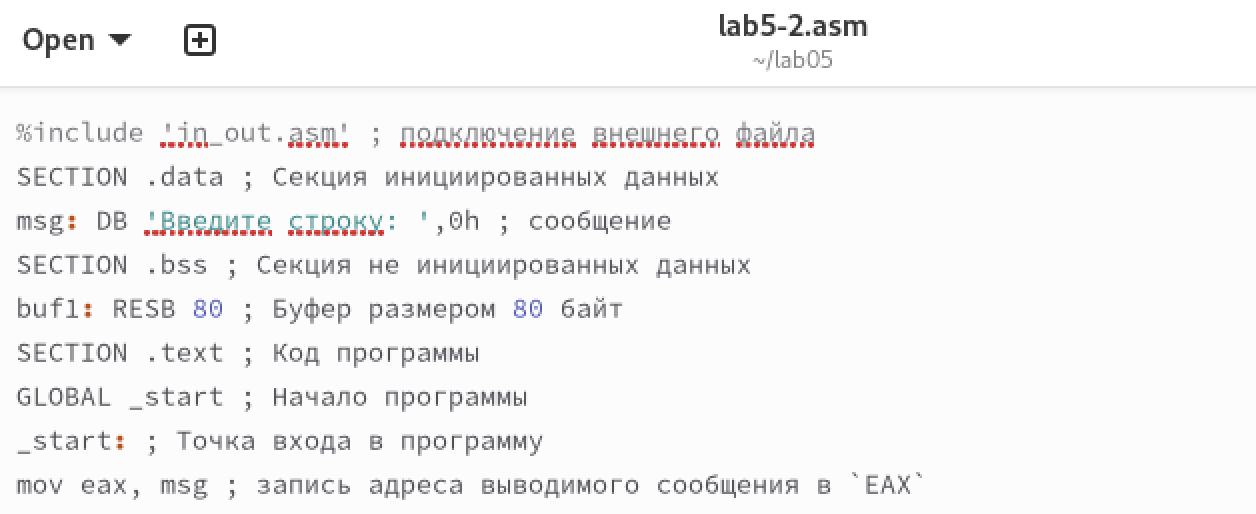
Копируем файл в нужную директорию

Теперь скопируем lab5-1 в ту же папку, назвав его lab5-2. (Рис.10)



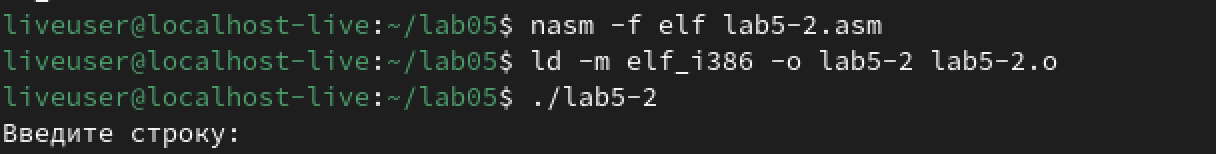
Копируем файл lab5-1 и переименовываем

Откроем новый файл и отредактируем его, вставив туда следующий код. (Рис. 11)



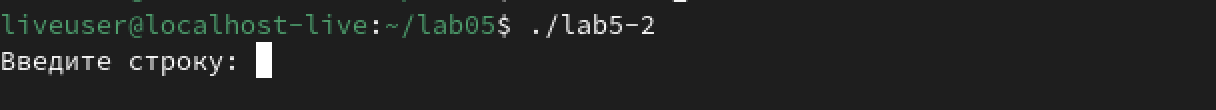
Редактируем файл

Теперь проделаем то же самое, что и с первым файлом, и запустим его. (Рис. 12)



Компилируем и запускаем файл

Он работает также, как и файл lab5-1, но использует для работы сторонний файл. Теперь отредактируем код еще раз и заменим sprintFL на sprint. Запускаем код. (Рис.13)

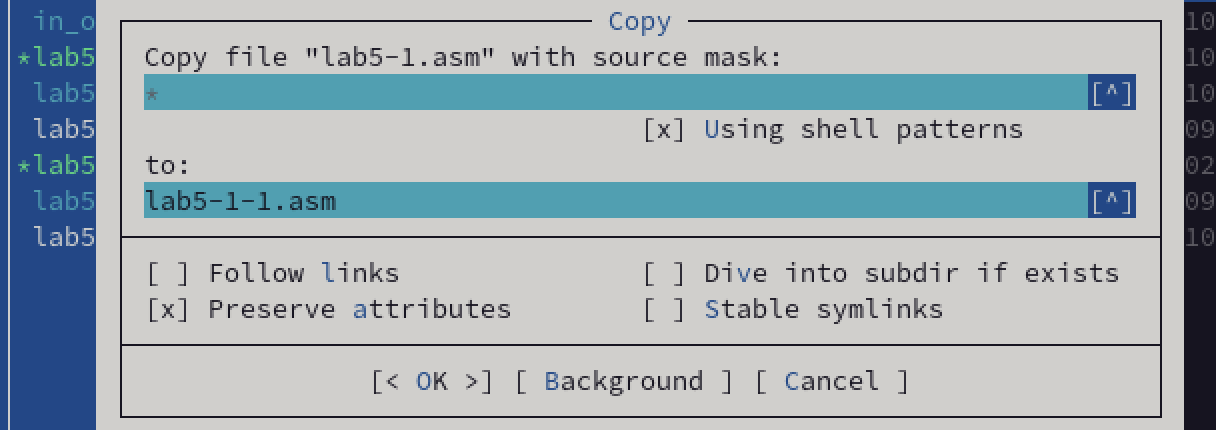


Запускаем код после редактуры

Нетрудно заметить, что теперь нет переноса на следующую строку. В этом и отлиичие команды sprintLF от sprint. Первая добавляет перенос после текста, а вторая нет.

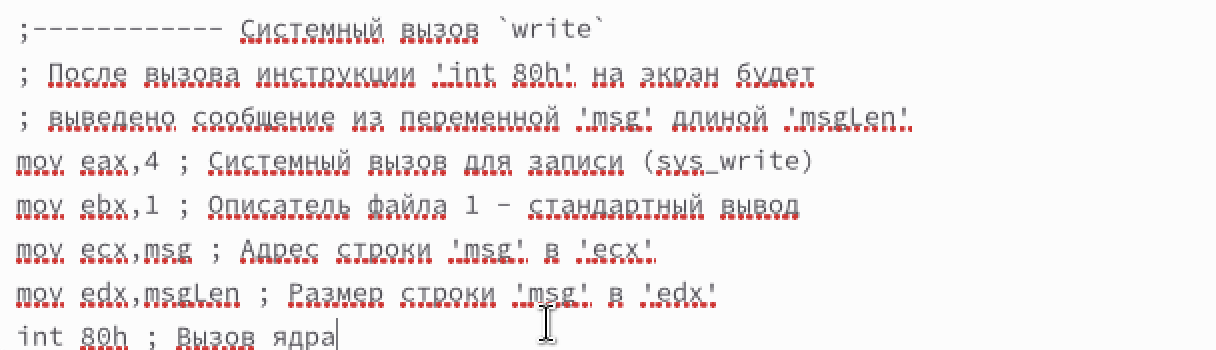
# 3 Выполнение самостоятельной работы

Теперь еще раз скопируем изначальный файл. (Рис.14)



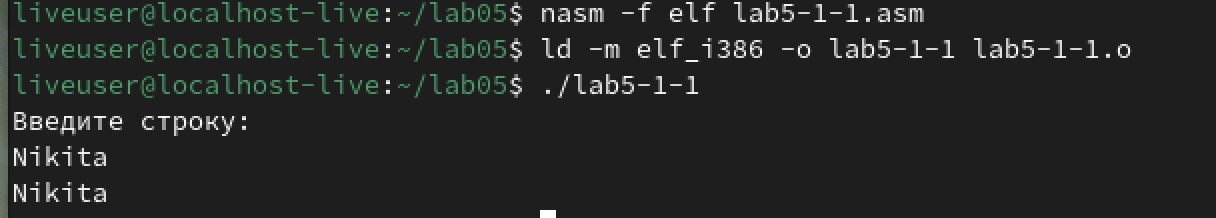
Копируем изначальный файл

Нам нужно сделать так,чтобы код выводил полученную на вводе строку. Для этого перед системным вызовом exit нам нужно вставить часть кода с вызовом write. Нам необходимо переместить адрес строки buf1 в ecx и размер строки buf1 (80) в edx. (Рис.15)



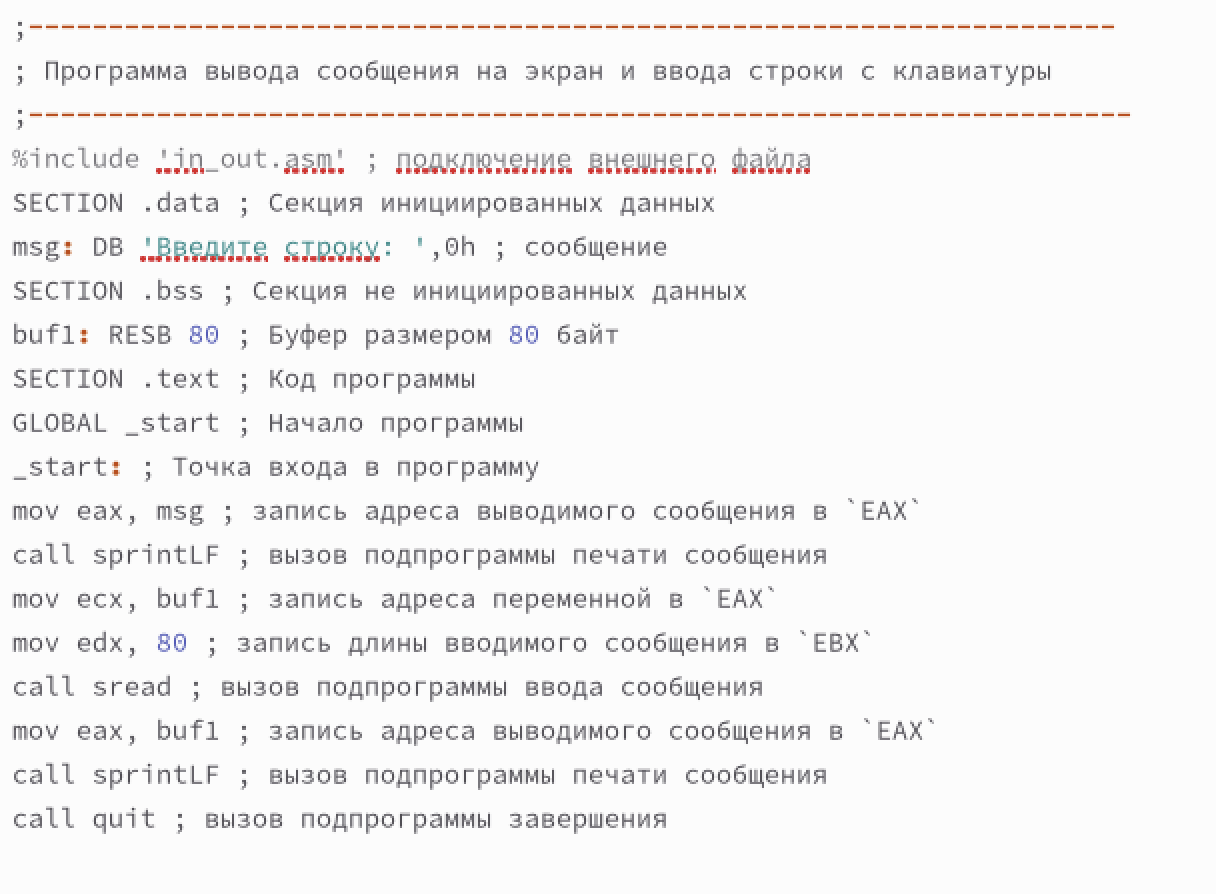
Редактируем файл, чтобы на выводе мы получали то же, что и вводили

Запускаем файл и проверяем работоспособность. (Рис.16)



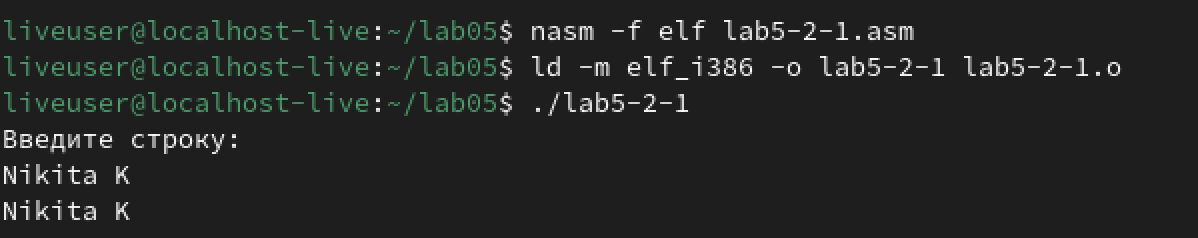
Запускаем файл

Теперь создадим копию файла lab5-2.asm и сделаем так, чтобы код тоже выводил тот же текст, что получает на ввод. Для этого перед последней строкой добавим строчу, которая записывает в eax адрес buf1 и строку, вызывающую подпрограмму sprintLF. (Рис.17)



Копируем второй файл и пишем код для вывода того же текста, что и при вводе

Скомпилируем файл и запустим код.(Рис.18)



Проверяем работоспособность кода

Программа выполняется верно, следовательно, самостоятельная работа выполнена правильно.

# 4 Выводы

В ходе выполнения данной лабораторной работы были получены навыки работы с Midnight commander и навыки создания программ ввода и вывода на ассемблере.