Лабораторная работа №5

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

Самарханова Полина Тимуровна

Содержание

Список иллюстраций

Список таблиц

1 Цель работы

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

2 Задание

Основы работы с NASM

Структура программы на языке ассемблера NASM

Подключение внешнего файла

Выполнение заданий для самостоятельной работы

3 Теоретическое введение

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). 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: • DB (define byte) — определяет переменную размером в 1 байт; • DW (define word) — определяет переменную размеров в 2 байта (слово); • DD (define double word) — определяет переменную размером в 4 байта (двойное слово); • DQ (define quad word) — определяет переменную размером в 8 байт (учетверённое слово); • DT (define ten bytes) — определяет переменную размером в 10 байт.

4 Выполнение лабораторной работы

4.1 Основы работы с NASM.

Я открыла Midnight Commander, используя команду mc(рис. [4.1]).

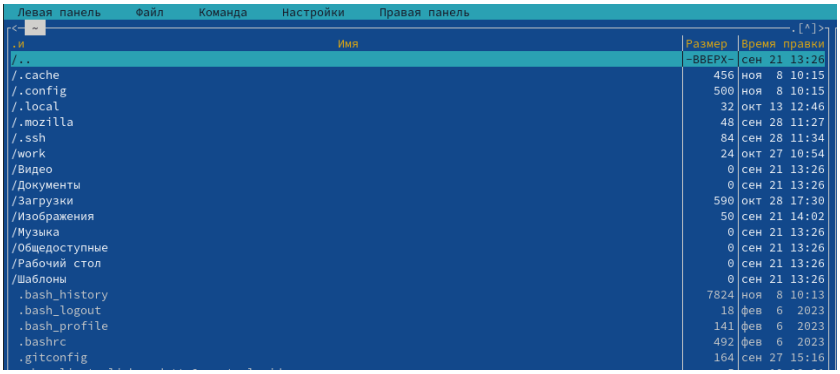


Рис. 4.1: Открытый Midnight Commander

Перешла в каталог work,созданный при выполнении предыдущей лабораторной работы(рис. [4.2]).

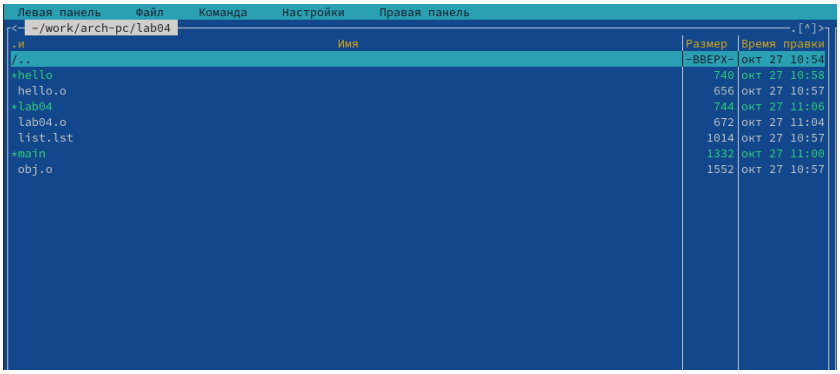


Рис. 4.2: Перемещение в каталог

Далее создала новый каталог lab05(рис. [4.3]).

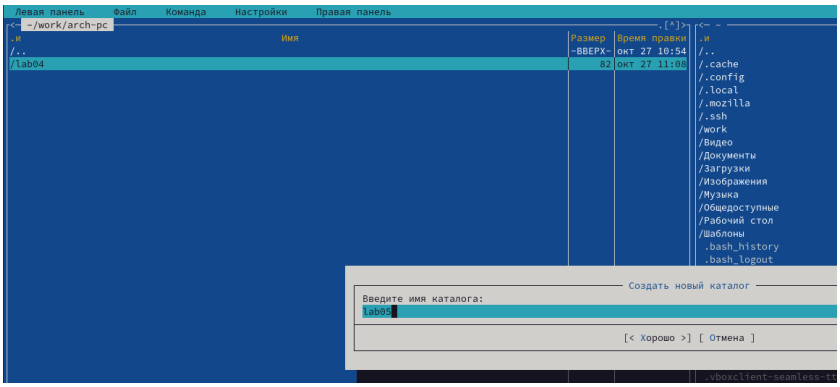


Рис. 4.3: Создание каталога

В новом каталоге создала файл lab5-1.asm,в котором я буду работать далее, используя команду touch(рис. [4.4]).

```
[spolina@fedora lab05]$ touch lab5-1.asm
```

Рис. 4.4: Создание файла

4.2 Структура программы на языке ассемблера NASM.

С помощью клавиши F4 я открыла созданный файл в редакторе nano(рис. [4.5]).

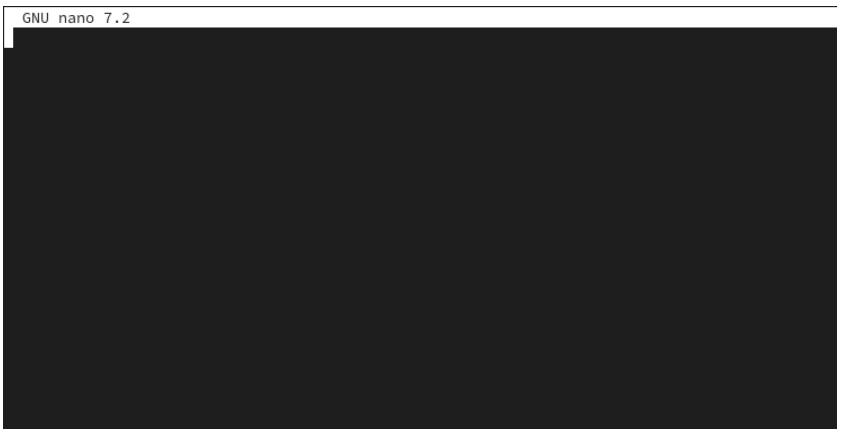


Рис. 4.5: Файл в редакторе

Я ввела в файл код программы для запроса строки(рис. [4.6]). Далее вышла из редактора,сохраняя изменения.

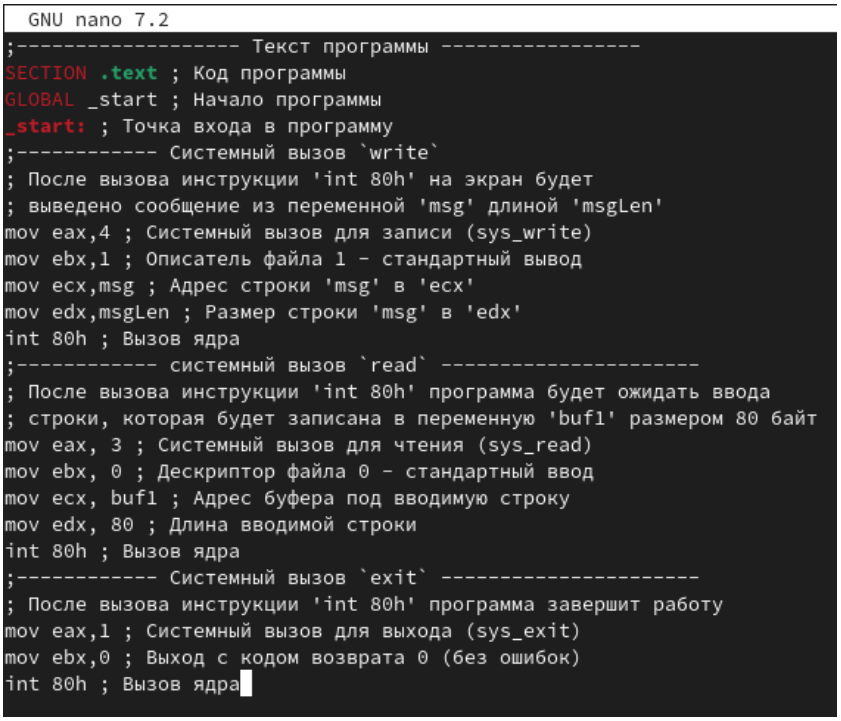


Рис. 4.6: Редактирование файла

Используя клавишу F4, я открыла файл для просмотра,чтобы проверить,сохранилась ли в нем написанная программа(рис. [4.7]).

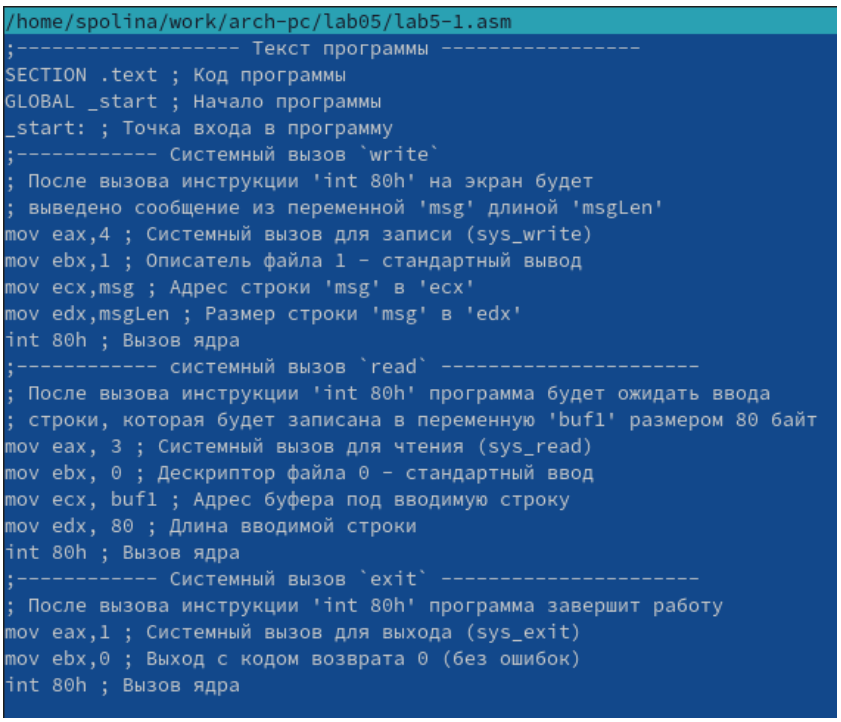


Рис. 4.7: Открытие файла для проверки

После этого я создала для текста программы объектный файл. Выполняю его компоновку(рис. [4.8]) (рис. [4.9]). После чего создался исполняемый файл lab5-1.

Рис. 4.8: Создание объектного файла

Рис. 4.9: Компоновка

Я запустила исполняемый файл. Программа выводит строку и ждет ввода с клавиатуры, после ввода своего ФИО программа завершает работу(рис. [4.10]).

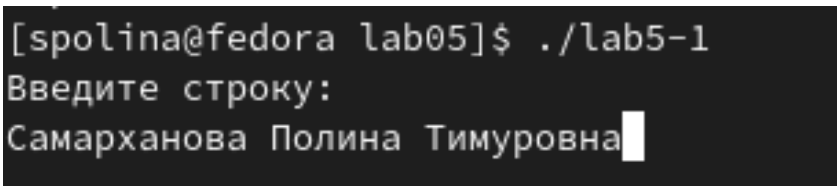


Рис. 4.10: Запуск программы

4.3 Подключенние внешнего файла.

Скачиваю файл in_out.asm со страницы ТУИС. Файл сохранился в ‘Загрузки’(рис. [4.11]).

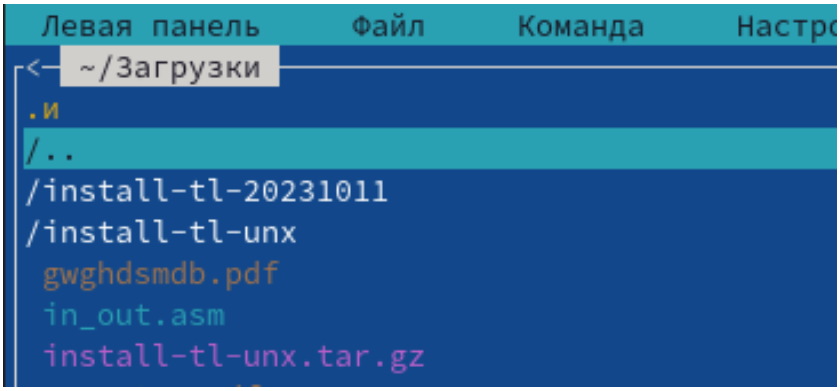


Рис. 4.11: Файл in_out.asm

Далее я копирую данный файл в каталог lab05, используя клавишу F5(рис. [4.12]).

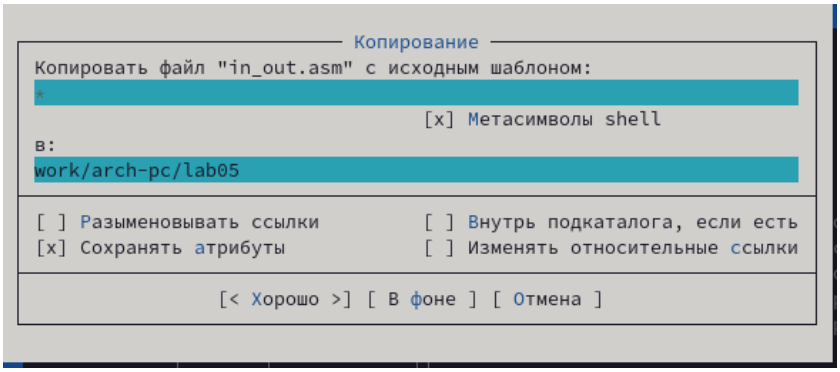
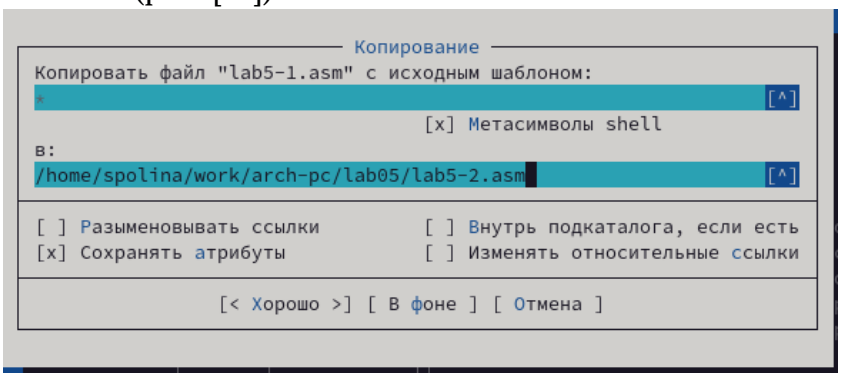


Рис. 4.12: Копирование файла в нужную директорию

С помощью той же утилиты F5 копировала файл lab5-1.asm, но уже с другим названием(рис. [**??**]).



После чего поменяла содержимое файла lab5-2.asm в редакторе nano, чтобы в программе использовались подпрограммы из внешнего файла in_out.asm(рис. [4.13]).

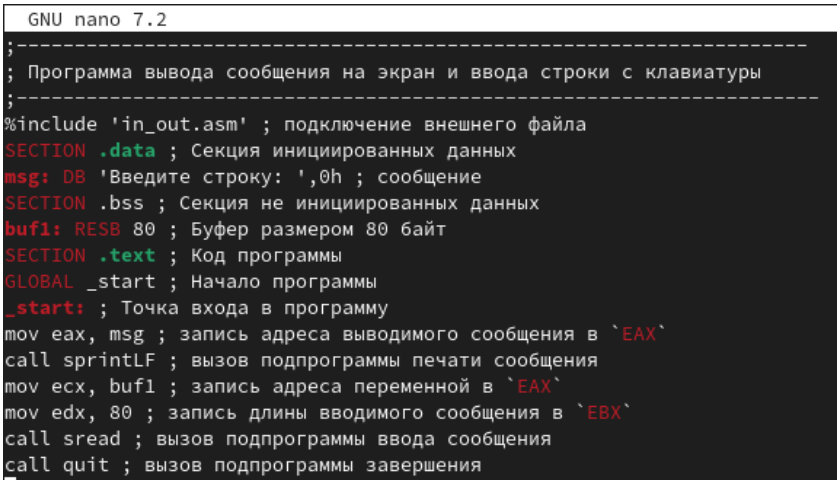


Рис. 4.13: Редактирование файла,для использования in_put.asm

Я создала объектный файл для lab5-2.asm(рис. [4.14]).

Рис. 4.14: Создание объектного файла

Я компоную данный файл,после чего создается исполняемый файл. Запустила его и проверила,работает ли данная программа(рис. [4.15]).

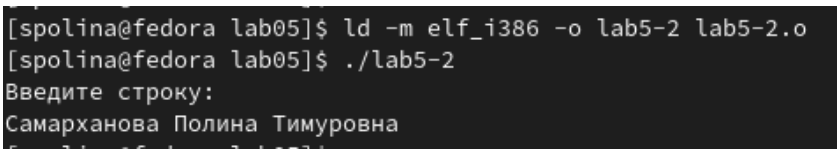


Рис. 4.15: Компоновка файла и запуск программы

Далее я открыла файл lab5-2.asm для редактирования в nano,используя F4. Изменила в нем подпрограмму sprintLF на sprint, сохранила изменения и открыла файл для проверки(рис. [4.16]).

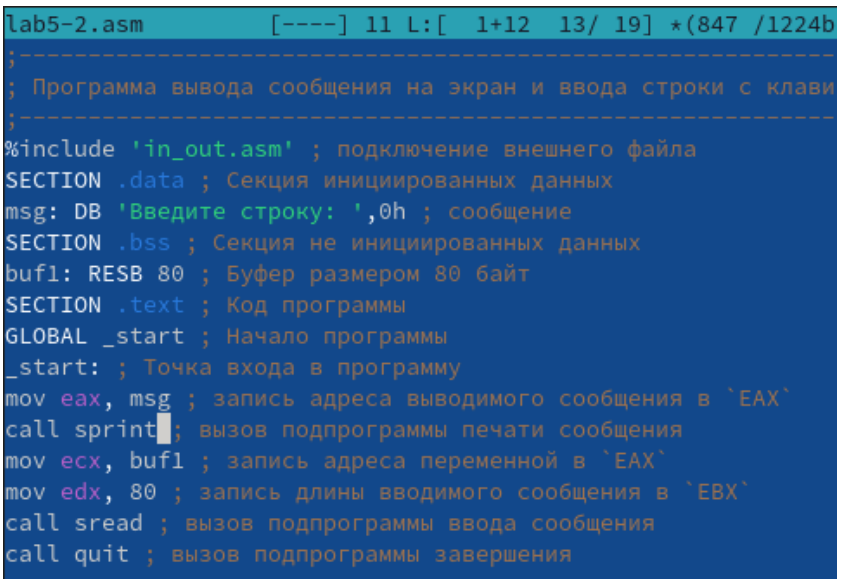


Рис. 4.16: Редактирование файла

После я делаю компоновку объектного файла и запускаю новый исполняемый файл(рис. [4.17]).

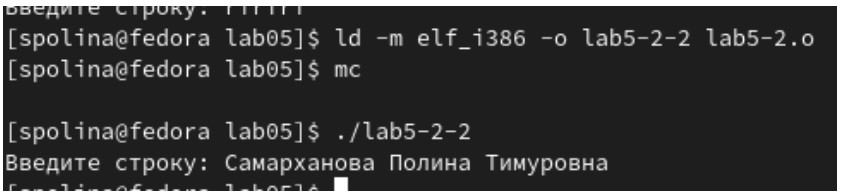


Рис. 4.17: Исполнение файла

Вся разница заключается в том, что запуск с подпрограммой sprintLF запрашивает воод с новой строки, а исполняемый файл с подпрограммой sprint просит ввод без переноса но новую строчку.

4.4 Выполнение заданий для самостоятельной работы.

Я создаюла копию файла lab5-1.asm с именем lab5-1-1.asm с помощью клавиши F5(рис. [4.18]).

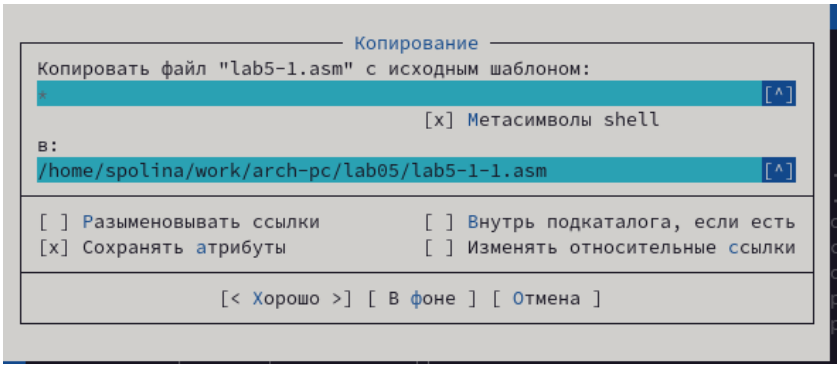


Рис. 4.18: Копирование файла lab5-1.asm

Используя клавишу F4 я открыла данный файл в nano и редактировала файл так,чтобы кроме вывода приглашения и запроса ввода, она выводила строку,которую пользователь ввел с клавиатуры(рис. [4.19]).

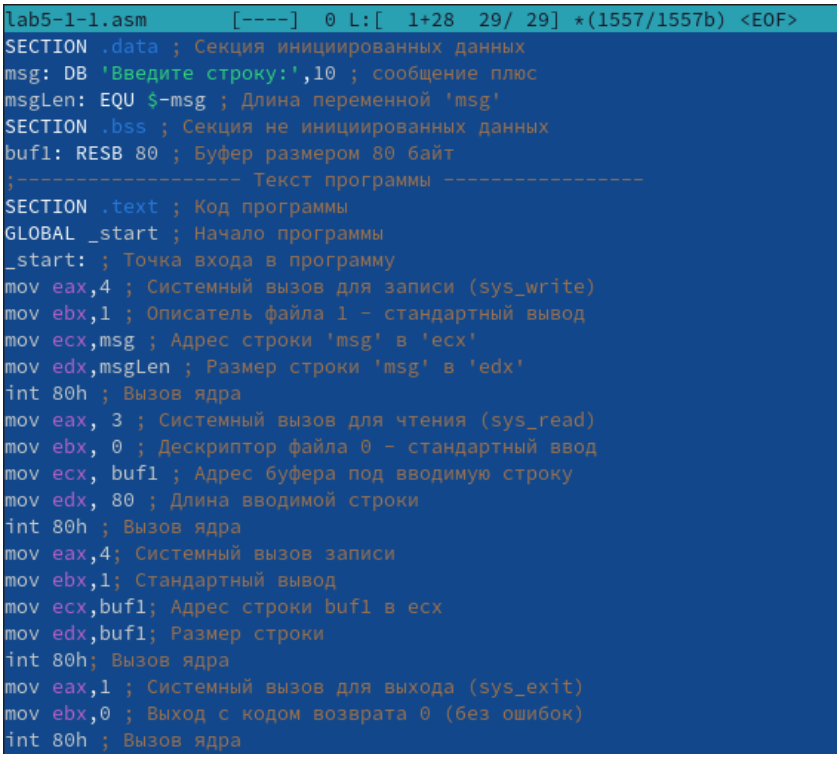


Рис. 4.19: Редактирование программы

Далее я создала объектный файл lab5-1-1.o и обработала его,используя

компоновщик,запустила созданный исполняемый файл, ввела своё имя. После этого программа вывела то,что я напечатал(рис. [4.20]).

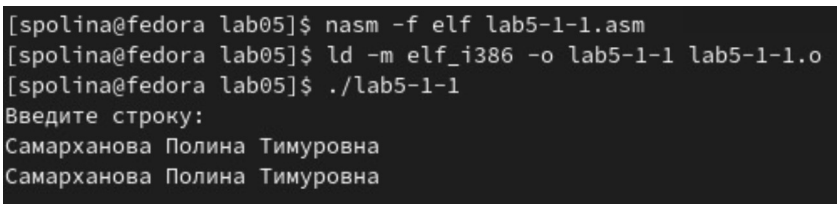


Рис. 4.20: Исполнение файла

После я скопировала файл lab5-2.asm, используя F5, переименовала его в lab5-2-3.asm(рис. [4.21]).

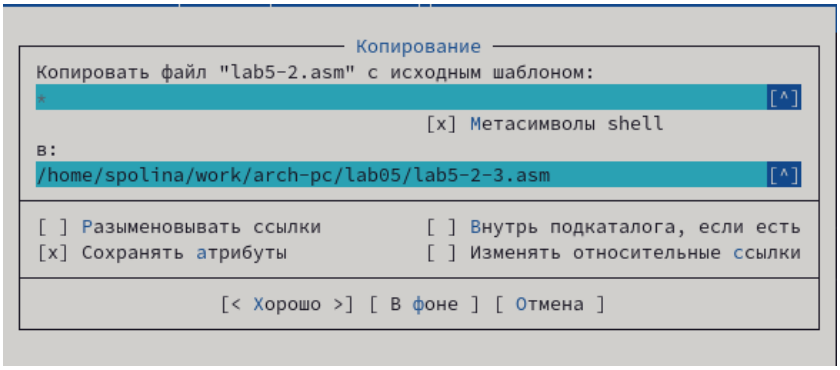


Рис. 4.21: Копирование файла

Используя клавишу F4 я открыла данный файл в nano и отредактировала файл так,чтобы кроме вывода приглашения и запроса ввода, она выводила строку,которую пользователь вводит с клавиатуры(рис. [4.22]).

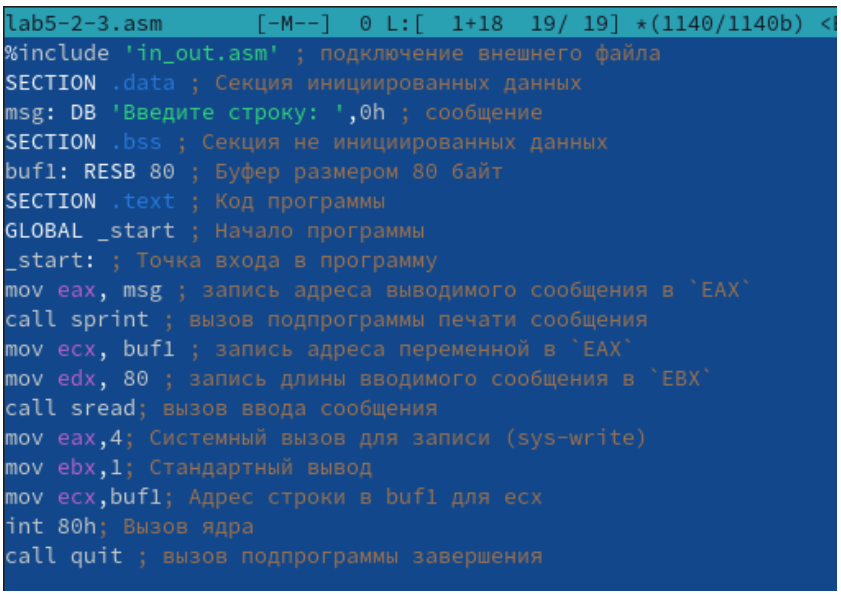


Рис. 4.22: Редактирование программы

И я создала объектный файл lab5-2-3.o и обработала его,используя компоновщик,запускаю созданный исполняемый файл, ввожу своё имя, после этого программа выводит то,что я напечатал(рис. [4.23]).

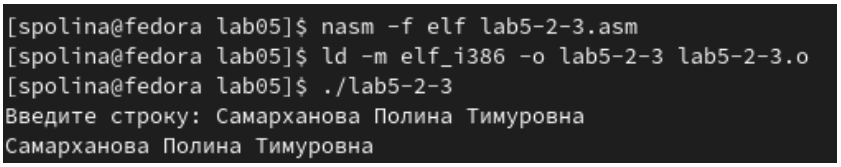


Рис. 4.23: Исполнение файла

5 Выводы

При выполнении данной работы я приобрела навыки работы с Midnight Commander, а также освоила инструкции языка ассемблера mov и int.

Список литературы

::: {#Лабораторная работа №5} :::