Лабораторная работа №5

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

Баранов Никита Дмитриевич

Содержание

# 1 Цель работы

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

# 2 Задание

Написать 2 программы по примеру и впоследствии изменить их по условию.

# 3 Выполнение лабораторной работы

Откройте Midnight Commander.Пользуясь клавишами ↑ , ↓ и Enter перейдите в каталог ~/work/arch-pc созданный при выполнении лабораторной работы №4.С помощью функциональной клавиши F7 создайте папку lab05 (рис. 5.3) и перейдите в созданный каталог.Пользуясь строкой ввода и командой touch создайте файл lab5-1.asm (рис. fig. 1)(рис. fig. 2)(рис. fig. 3).

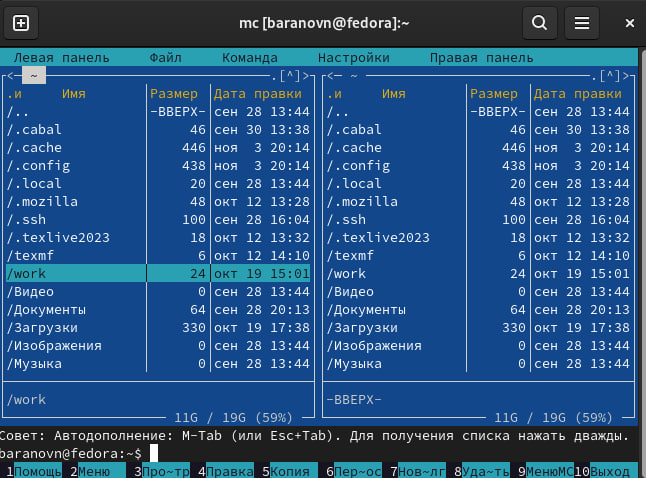


Рис. 1: Используем команду mc

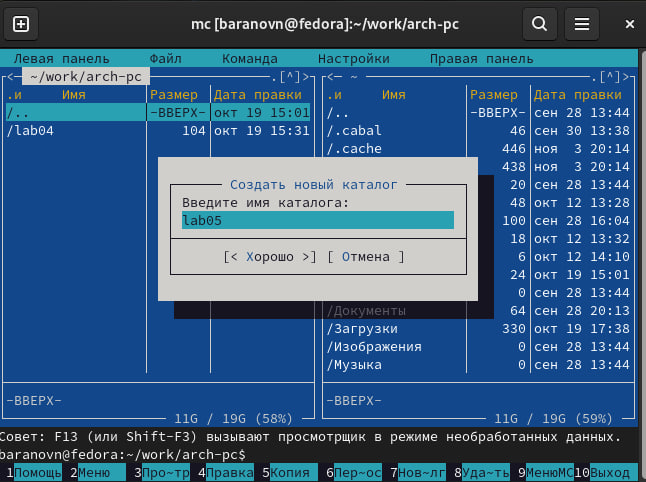


Рис. 2: Создаем папку lab05

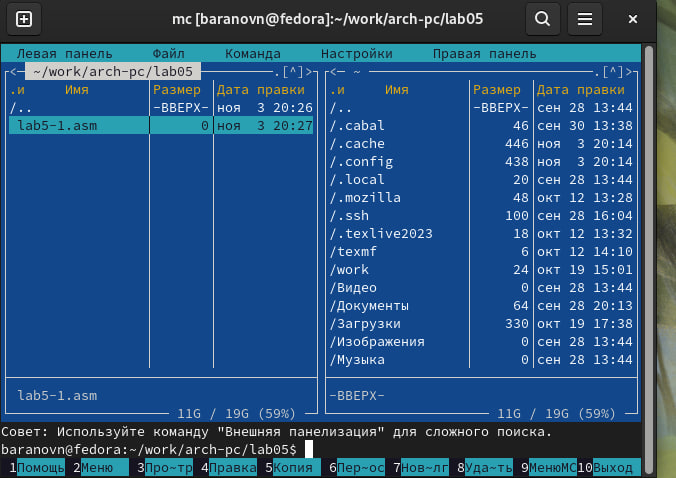


Рис. 3: С помощью команды touch создаем файл

С помощью функциональной клавиши F4 откройте файл lab5-1.asm для редактирования во встроенном редакторе. Как правило в качестве встроенного редактора Midnight Commander используется редакторы nano (рис. 5.5) или mcedit.Введите текст программы из листинга 5.1 (можно без комментариев), сохраните изменения и закройте файл. Оттранслируйте текст программы lab5-1.asm в объектный файл. Выполните компоновку объектного файла и запустите получившийся исполняемый файл(рис. fig. 4)(рис. fig. 5).

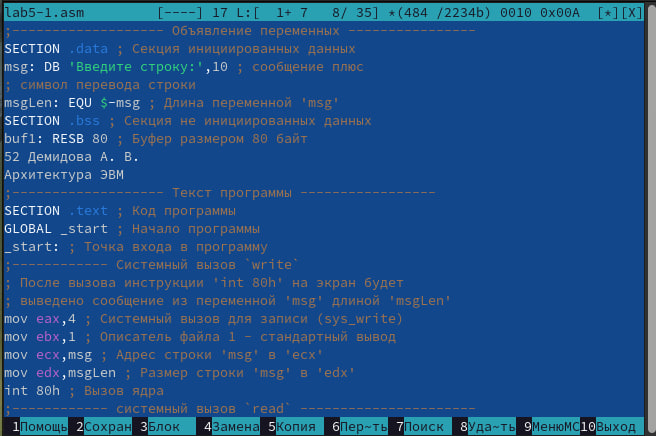


Рис. 4: Редактируем файл

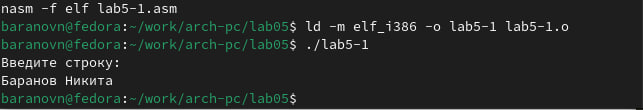


Рис. 5: Оттранслируем текст программы в объектный файл и запустим файл

Скачайте файл in\_out.asm. В одной из панелей mc откройте каталог с файлом lab5-1.asm. В другой панели каталог со скаченным файлом in\_out.asm (для перемещения между панелями используйте Tab ).Скопируйте файл in\_out.asm в каталог с файлом lab5-1.asm с помощью функциональной клавиши F5.(рис. fig. 6).

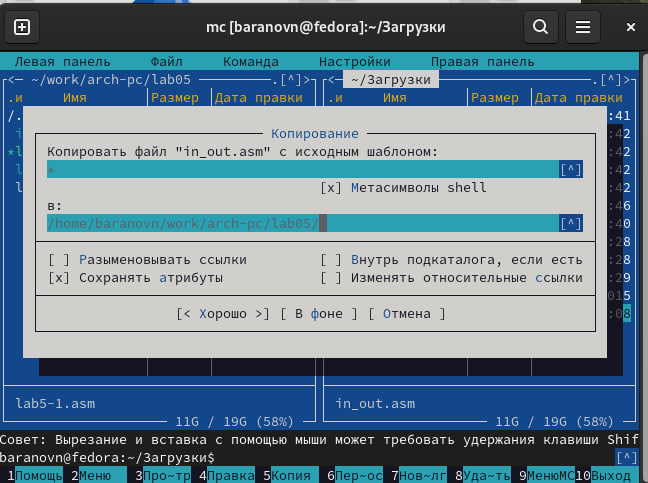


Рис. 6: Копируем файл в нужный каталог

С помощью функциональной клавиши F6 создайте копию файла lab5-1.asm с именем lab5-2.asm.Исправьте текст программы в файле lab5-2.asm с использование подпрограмм из внешнего файла in\_out.asm (используйте подпрограммы sprintLF, sread и quit) в соответствии с листингом 5.2. Создайте исполняемый файл и проверьте его работу.В файле lab5-2.asm замените подпрограмму sprintLF на sprint. Создайте исполняемый файл и проверьте его работу. В чем разница?(рис. fig. 7)(рис. fig. 8)(рис. fig. 9)(рис. fig. 10)(рис. fig. 11).

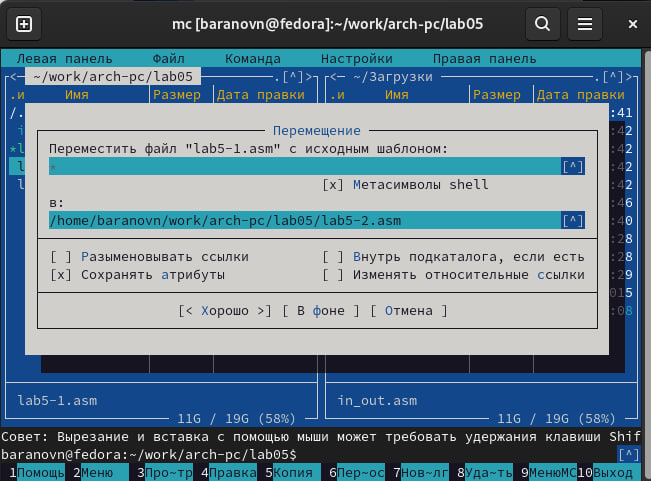


Рис. 7: Копируем файл, переименовывая его в lab5-2.asm

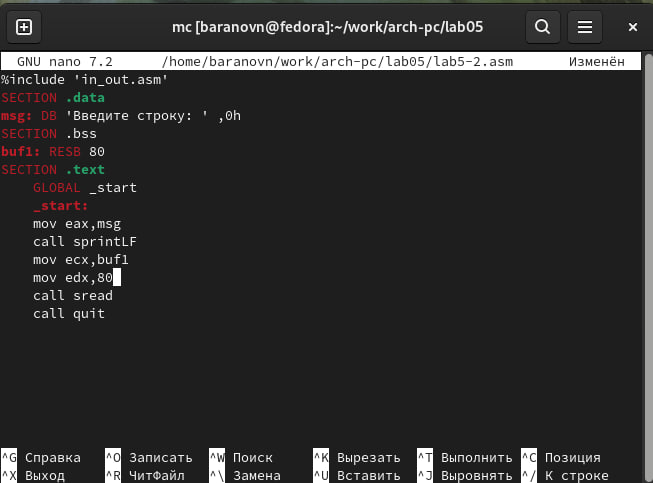


Рис. 8: Редактируем файл с использованием подпрограмм

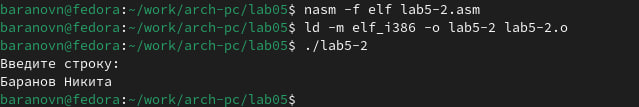


Рис. 9: Запускаем файл

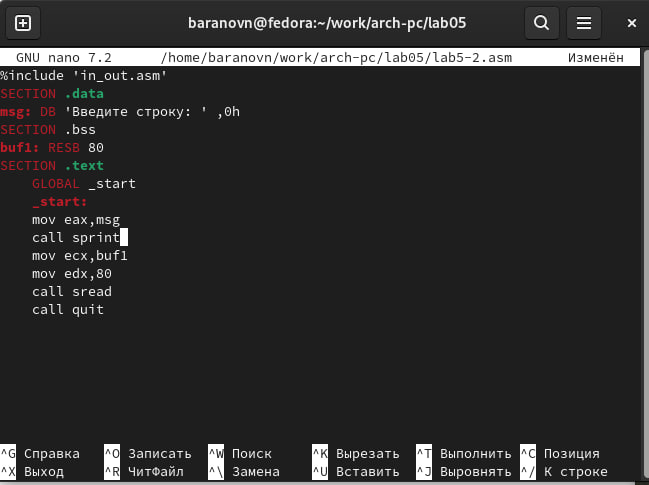


Рис. 10: Меняем подпрограммы

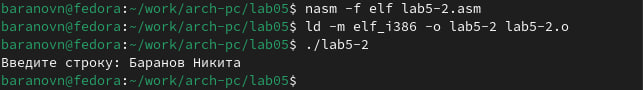


Рис. 11: Запускаем файл и сравниваем с прошлым результатом

Сравнивая два программы, я сделал вывод, что sprintLF переносит ввод на новую строку, тогда как sprint дает вводить строку без переноса.

# 4 Самостоятельная работа

Создайте копию файла lab5-1.asm. Внесите изменения в программу (без использования внешнего файла in\_out.asm), так чтобы она выводила строку которую вы ввели(рис. fig. 12)(рис. fig. 13)(рис. fig. 14).

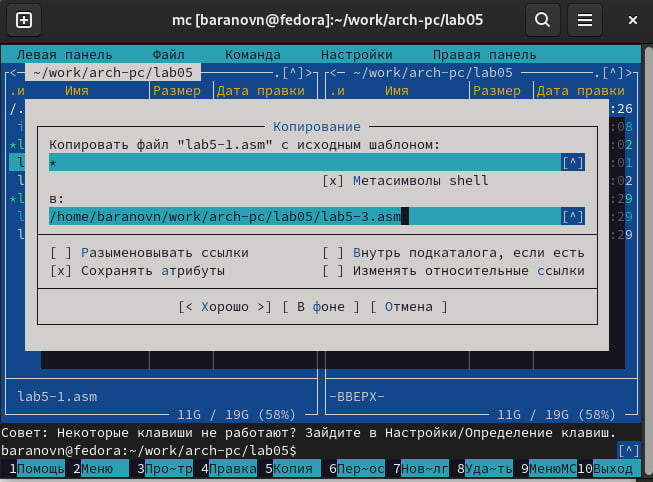


Рис. 12: Копируя, создаем новый файл lab5-3.asm

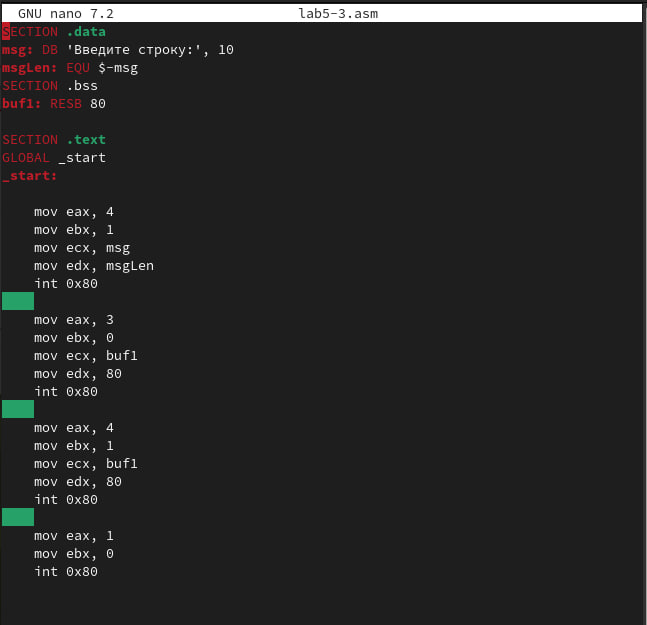


Рис. 13: Редактируем файл

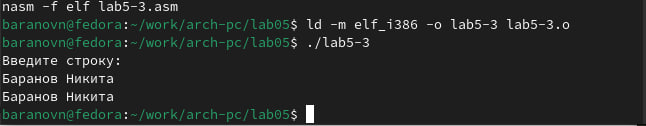


Рис. 14: Запускаем файл

Создайте копию файла lab5-2.asm. Исправьте текст программы с использование подпрограмм из внешнего файла in\_out.asm, так чтобы она работала по следующему алгоритму(рис. fig. 15)(рис. fig. 16)(рис. fig. 17).

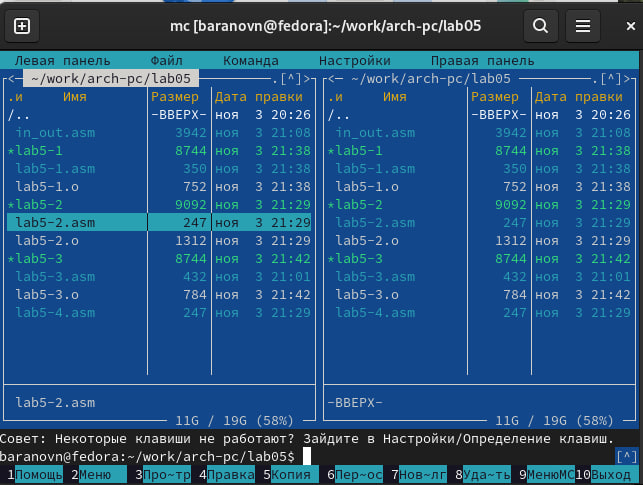


Рис. 15: Создаем новый файл, копируя прошлые

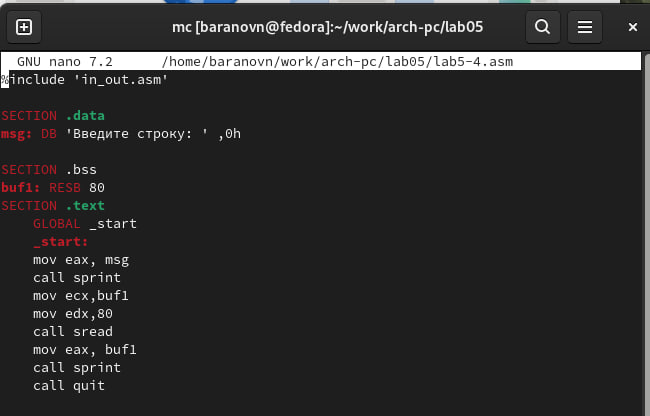


Рис. 16: Редактируем файл

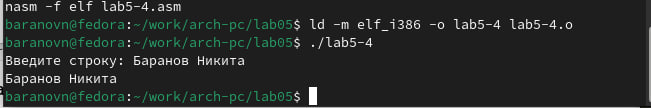


Рис. 17: Запускаем файл

# 5 Выводы

Мы приобрели навыки работы с Midnight Commander и освоили инструкцию mov.