Шаблон отчёта по лабораторной работе 5

Структура программы на языке ассемблера NASM. Системные вызовы в ОС GNU Linux

Чуева Злата Станиславовна

Содержание

1. Цель работы
2. Задание
3. Теоретическое введение
4. Выполнение лабораторной работы
5. Вывод

# Цель работы

Приобрести практические навыки работы в Midnight Commander. Освоить инструкции языка ассемблера mov и in.

# Задание

1. Открыть Midnight Commander
2. Создаь папку lab05 и файл lab5-1.asm в ней
3. В файл lab5-1.asm ввести информацию из листинга 5.1 и сохранить изменения
4. Оттранслиовать текст файла lab5-1.asm и выполнить компановку объектного файла
5. Запустить файл
6. Скачать файл in\_out.asm и скопировать его в каталог lab05
7. С помощью клавиши F6 создать копию файла lab5-1.asm с именем lab5-2.asm
8. Исправить текст программы в файле lab5-2.asm в соответствии с листингом 5.2
9. В файле lab5-2.asm заменить подпрограмму sprintLF на sprint
10. Создать исполняемый файл и проверить его работу
11. Создать копию файла lab5-1.asm и внести изменения, что бы программа выводила введенную строку
12. Создать копию файла lab5-2.asm и внести изменения, что бы программа выводила введенную строку

# Теоретическое введение

Простейший диалог с пользователем требует наличия двух функций — вывода текста на экран и ввода текста с клавиатуры. Простейший способ вывести строку на экран — использо- вать системный вызов write. Этот системный вызов имеет номер 4, поэтому перед вызовом инструкции int необходимо поместить значение 4 в регистр eax. Первым аргументом write, помещаемым в регистр ebx, задаётся дескриптор файла. Для вывода на экран в качестве дескриптора файла нужно указать 1 (это означает «стандартный вывод», т. е. вывод на экран). Вторым аргументом задаётся адрес выводимой строки (помещаем его в регистр ecx, напри- мер, инструкцией mov ecx, msg). Строка может иметь любую длину. Последним аргументом (т.е. в регистре edx) должна задаваться максимальная длина выводимой строки. Для ввода строки с клавиатуры можно использовать аналогичный системный вызов read. Его аргументы – такие же, как у вызова write, только для «чтения» с клавиатуры используется файловый дескриптор 0 (стандартный ввод). Системный вызов exit является обязательным в конце любой программы на языке ассем- блер. Для обозначения конца программы перед вызовом инструкции int 80h необходимо поместить в регистр еах значение 1, а в регистр ebx код завершения 0.

# Выполнение лабораторной работы

1. Открыть Midnight Commander
2. Создаь папку lab05 и файл lab5-1.asm в ней (смотреть рис.1)

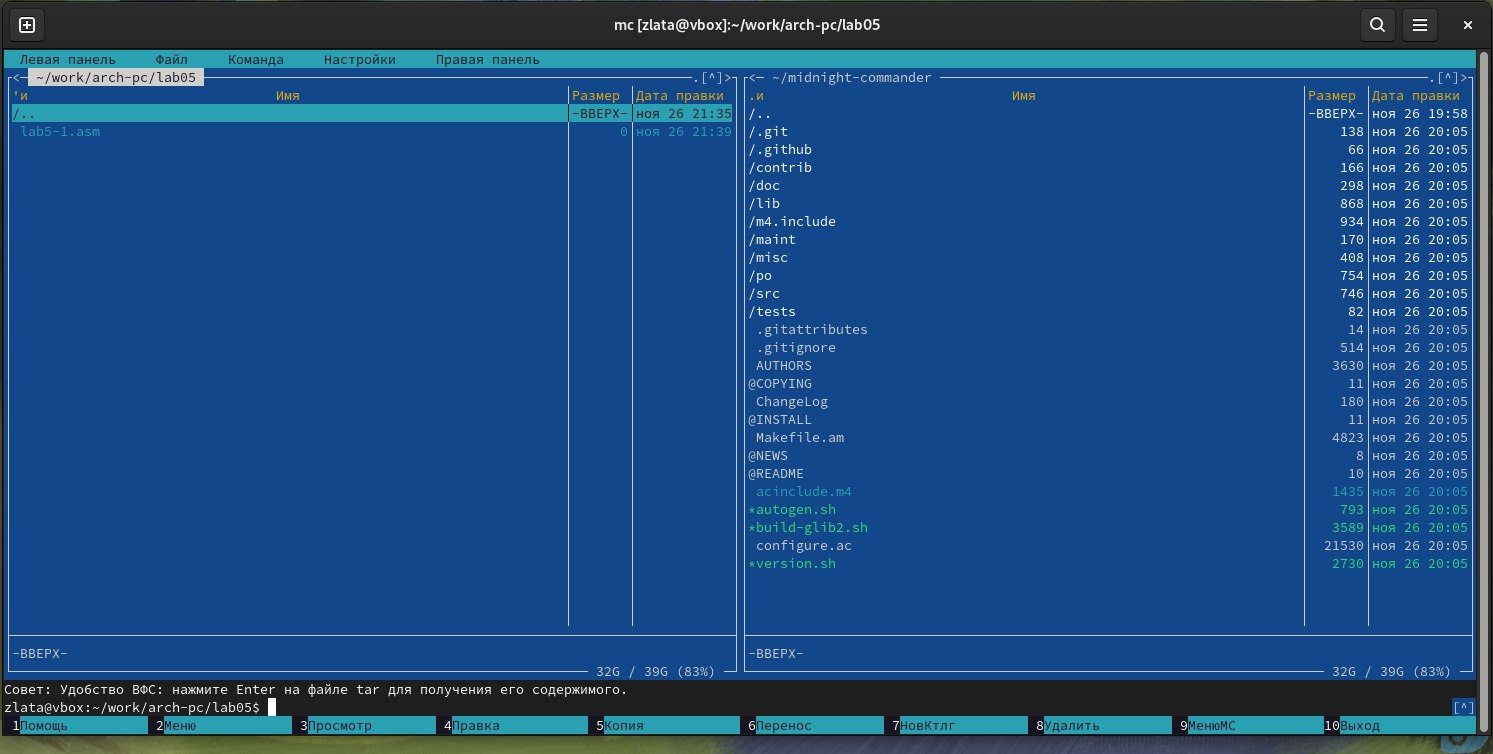


рисунок 1

1. В файл lab5-1.asm ввести информацию из листинга 5.1 и сохранить изменения (смотреть рис.2)

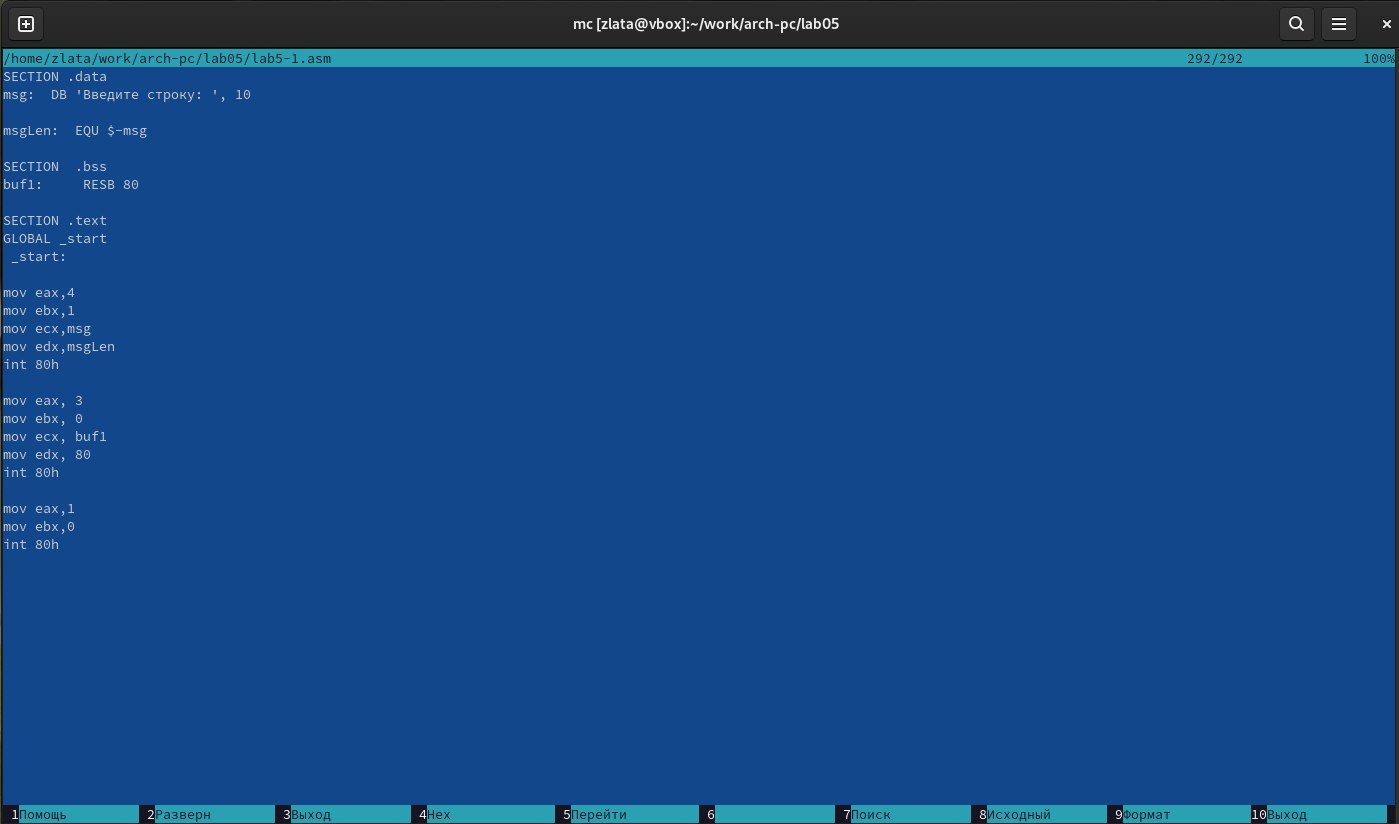


рисунок 2

1. Убедится что файл содержит информацию
2. Оттранслиовать текст файла lab5-1.asm и выполнить компановку объектного файла (смотреть рис. 3)
3. Запустить файл (смотреть рис. 3)

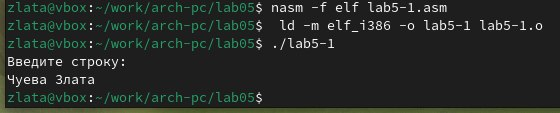


рисунок 3

1. Скачать файл in\_out.asm и скопировать его в каталог lab05 с помощью клавиши F5 (смотреть рис. 4)

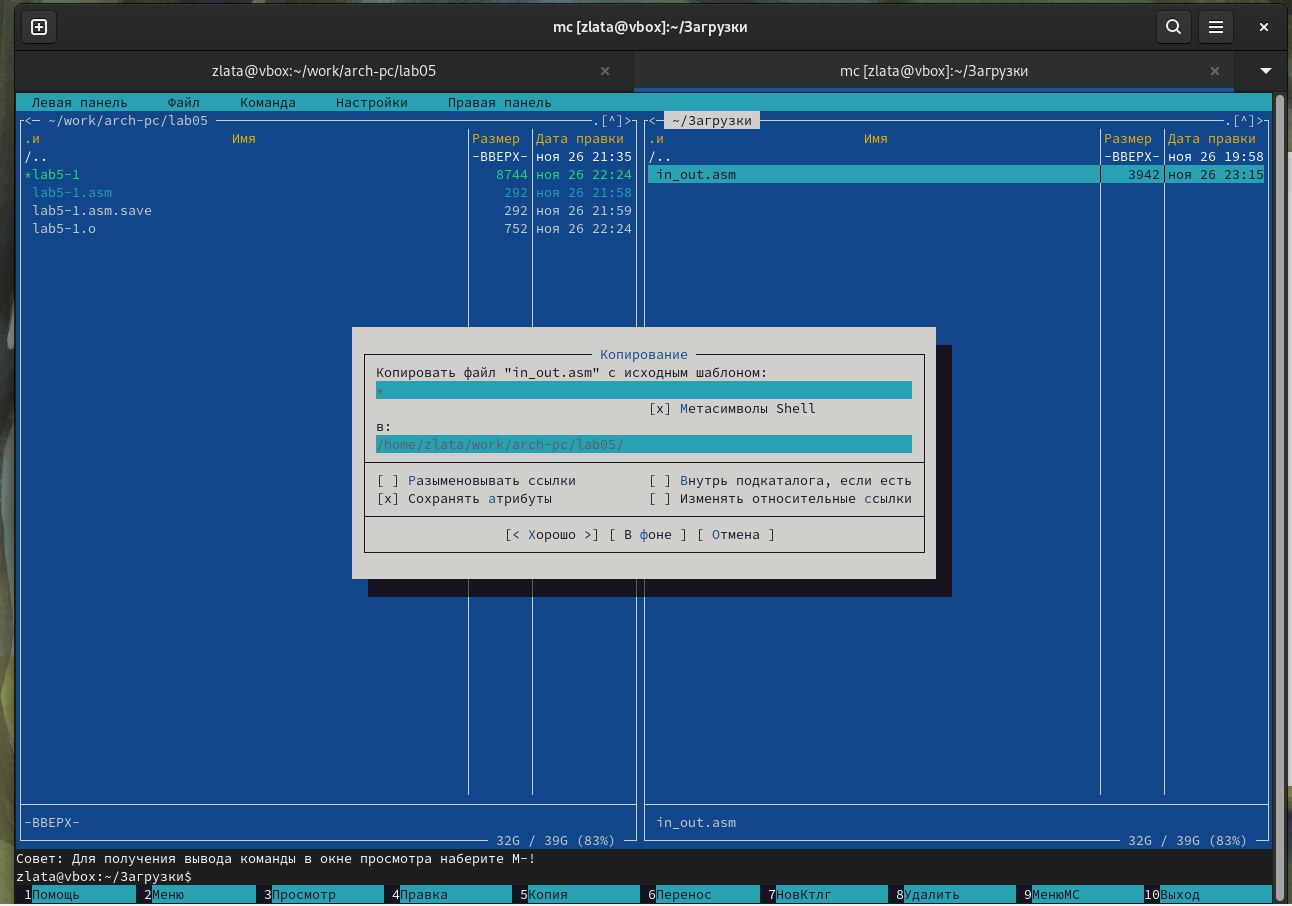


рисунок 4

1. С помощью клавиши F6 создать копию файла lab5-1.asm с именем lab5-2.asm (смотреть рис.5)

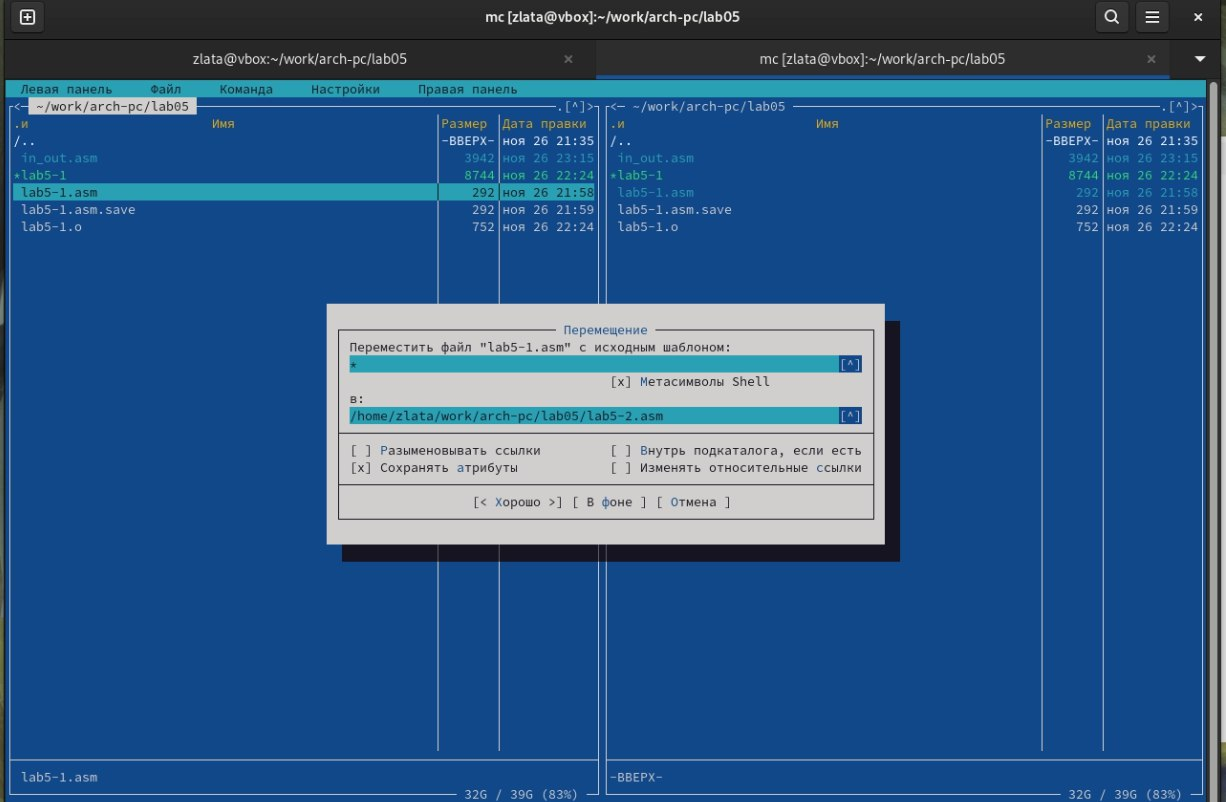


рисунок 5

1. Исправить текст программы в файле lab5-2.asm в соответствии с листингом 5.2 и заменить подпрограмму sprintLF на sprint (смотреть рис.6)

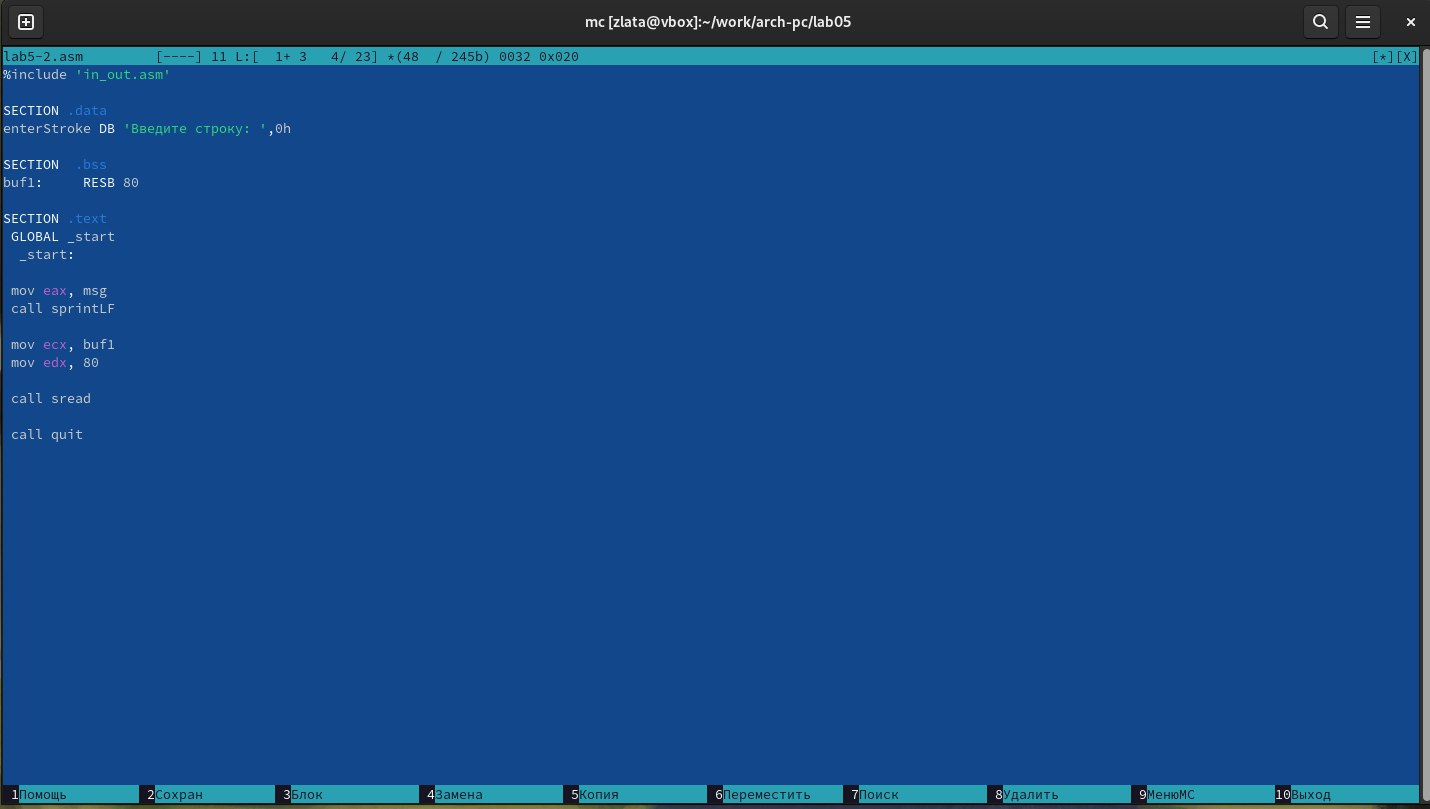


рисунок 6

1. Создать исполняемый файл и проверить его работу (смотреть рис.7)

рисунок 7

рисунок 7

(После вывода сообщения нет прехода на новую сроку)

1. Создать копию файла lab5-1.asm и внести изменения, что бы программа выводила введенную строку (смотреть рис.8)
2. Создать копию файла lab5-2.asm и внести изменения, что бы программа выводила введенную строку (смотреть рис.8)

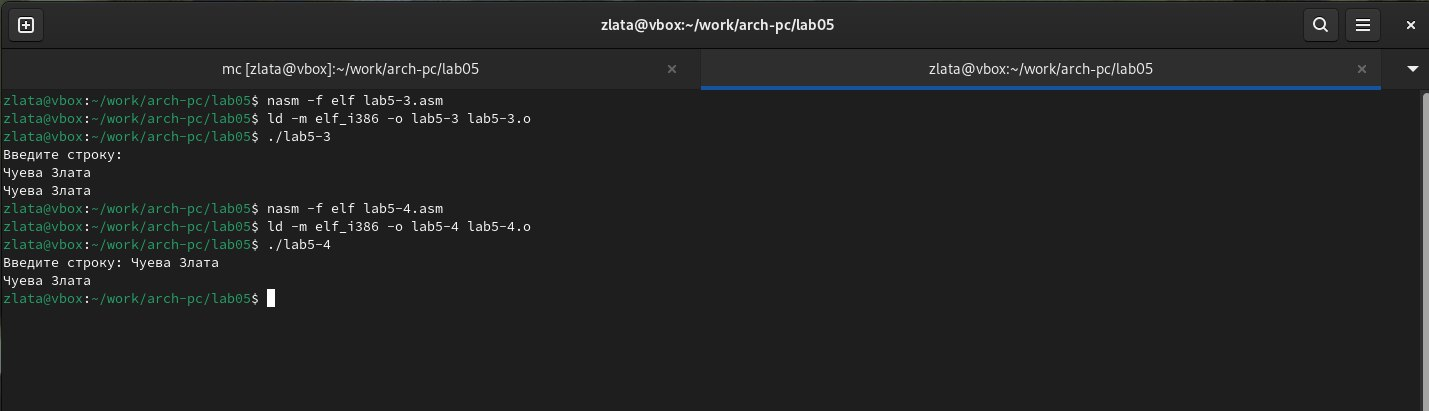


рисунок 8

# Вывод

Приобрела практические навыки работы в Midnight Commander. Освоила инструкции языка ассемблера mov и in.

# Список литературы