

Отчёт по лабораторной работе№5

Основы работы с Midnight Commander (mc). Структура программы на языке ассемблера NASM.

Шмавонян Диана Арамовна

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Выводы	14

Список иллюстраций

3.1	Открытие Midnight Commander	7
3.2	Создание папки для лабораторной работы	7
3.3	Папка lab05	8
3.4	Создание файла lab5-1.asm	8
3.5	Ввод текста	9
3.6	Транслирование текста, проверка работоспособности	9
3.7	Перенос файла в папку lab05	10
3.8	Создание копии	11
3.9	Внесение изменений в файл	11
3.10	Транслирование текста, проверка работоспособности	12
3.11	Проверка файла3	12
3.12	Проверка работоспособности	13

Список таблиц

1 Цель работы

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

2 Задание

1. Создайте копию файла lab6-1.asm. Внесите 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.
2. Получите исполняемый файл и проверьте его работу. На приглашение ввести строку введите свою фамилию.
3. Создайте копию файла lab6-2.asm. Исправьте текст программы с использование подпрограмм из 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.

3 Выполнение лабораторной работы

1. Создайте каталог для работы с программами на языке ассемблера NASM.

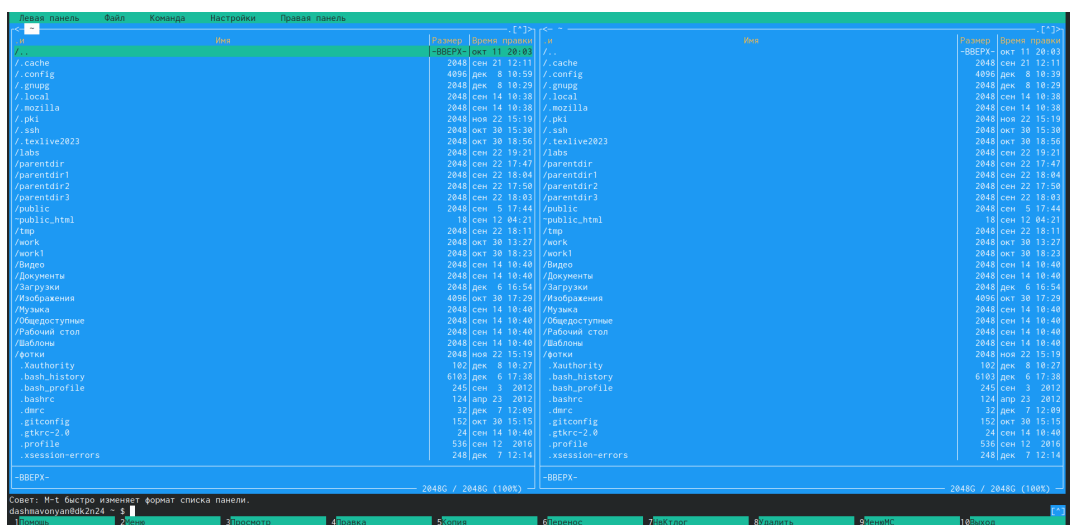


Рис. 3.1: Открытие Midnight Commander

2. С помощью функциональной клавиши F7 создаём папку lab05.

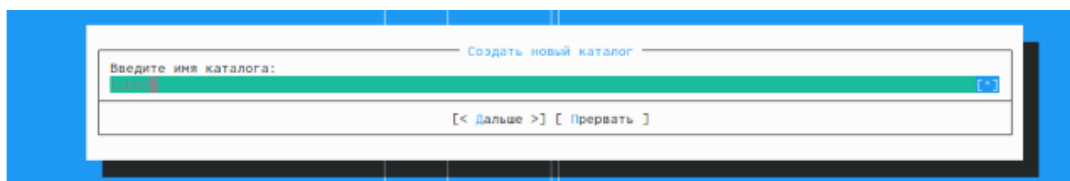


Рис. 3.2: Создание папки для лабораторной работы

3. Убедимся в правильном создании папки.

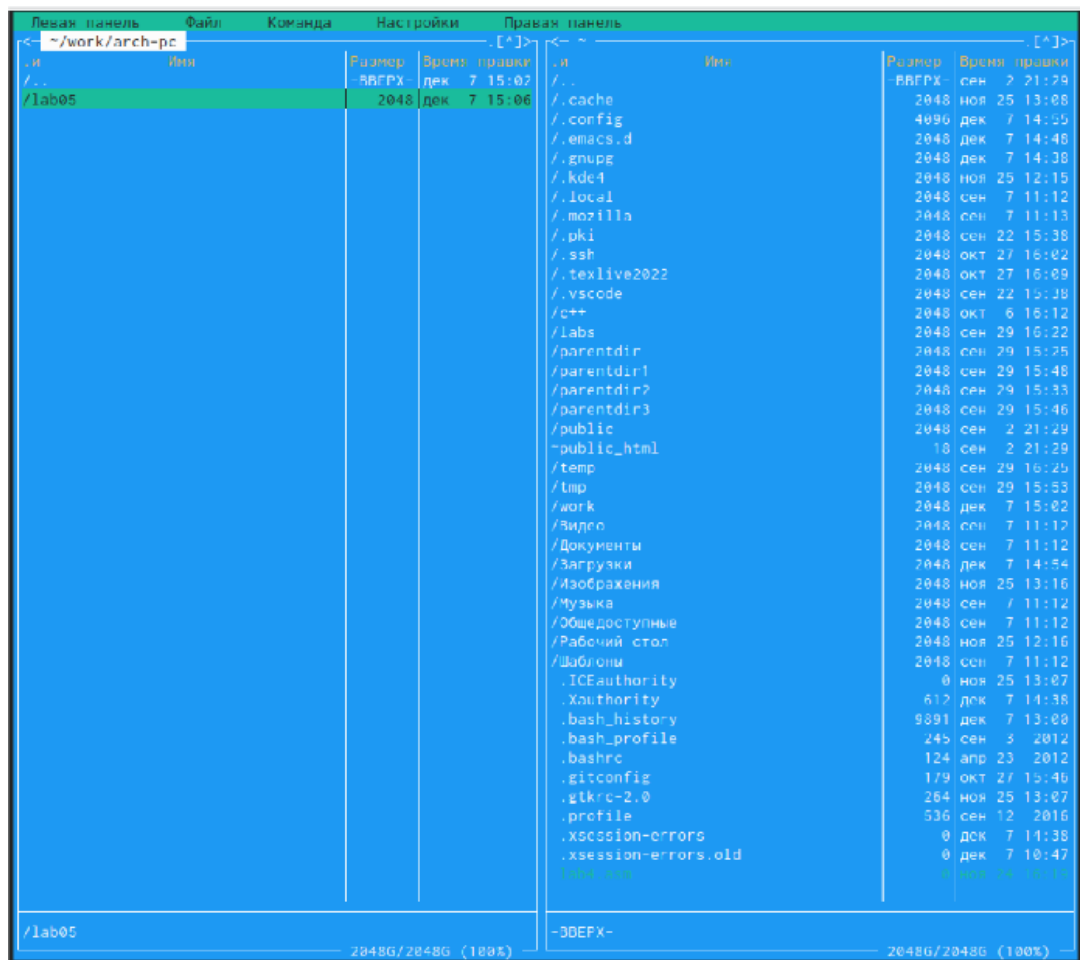


Рис. 3.3: Папка lab05

4. Пользуясь строкой ввода и командой touch создадим файл lab5-1.asm.

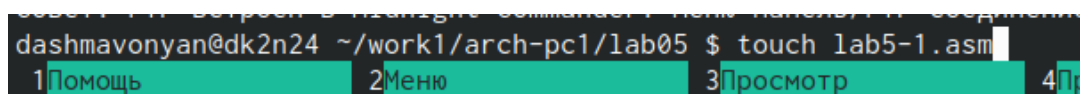


Рис. 3.4: Создание файла lab5-1.asm

5. С помощью функциональной клавиши F4 откроем файл lab5-1.asm и введём текст из листинга 6.1.


```

GNU nano 7.2 /afs/.dk.sci.pfu.edu.ru/home/d/a/dashmavonyan/work1/arch-pc1/lab05/lab5-1.asm
SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
; символ перевода строки
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
mov eax,4 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Описатель файла 1 - стандартный вывод
mov ecx,msg ; Адрес строки 'msg' в 'ecx'
mov edx,msgLen ; Размер строки 'msg' в 'edx'
int 80h ; Вызов ядра
mov eax, 3 ; Системный вызов для чтения (sys_read)
mov ebx, 0 ; Дескриптор файла 0 - стандартный ввод
mov ecx, buf1 ; Адрес буфера под вводимую строку
mov edx, 80 ; Длина вводимой строки
int 80h ; Вызов ядра
mov eax,1 ; Системный вызов для выхода (sys_exit)
mov ebx,0 ; Выход с кодом возврата 0 (без ошибок)
int 80h ; Вызов ядра

```

Рис. 3.5: Ввод текста

- Оттранслируем текст программы lab5-1.asm в объектный файл.

```

dashmavonyan@dk2n26 ~/work1/arch-pc1/lab05 $ mc

dashmavonyan@dk2n26 ~/work1/arch-pc1/lab05 $ ld -m elf_i386 -o lab5-1 lab5-1.o
dashmavonyan@dk2n26 ~/work1/arch-pc1/lab05 $ ./lab5-1
Введите строку:
Шмавонян Диана
dashmavonyan@dk2n26 ~/work1/arch-pc1/lab05 $

```

Рис. 3.6: Транслирование текста, проверка работоспособности

- Скопируем файл in_out.asm в каталог с файлом lab5-1.asm с помощью функциональной клавиши F5.

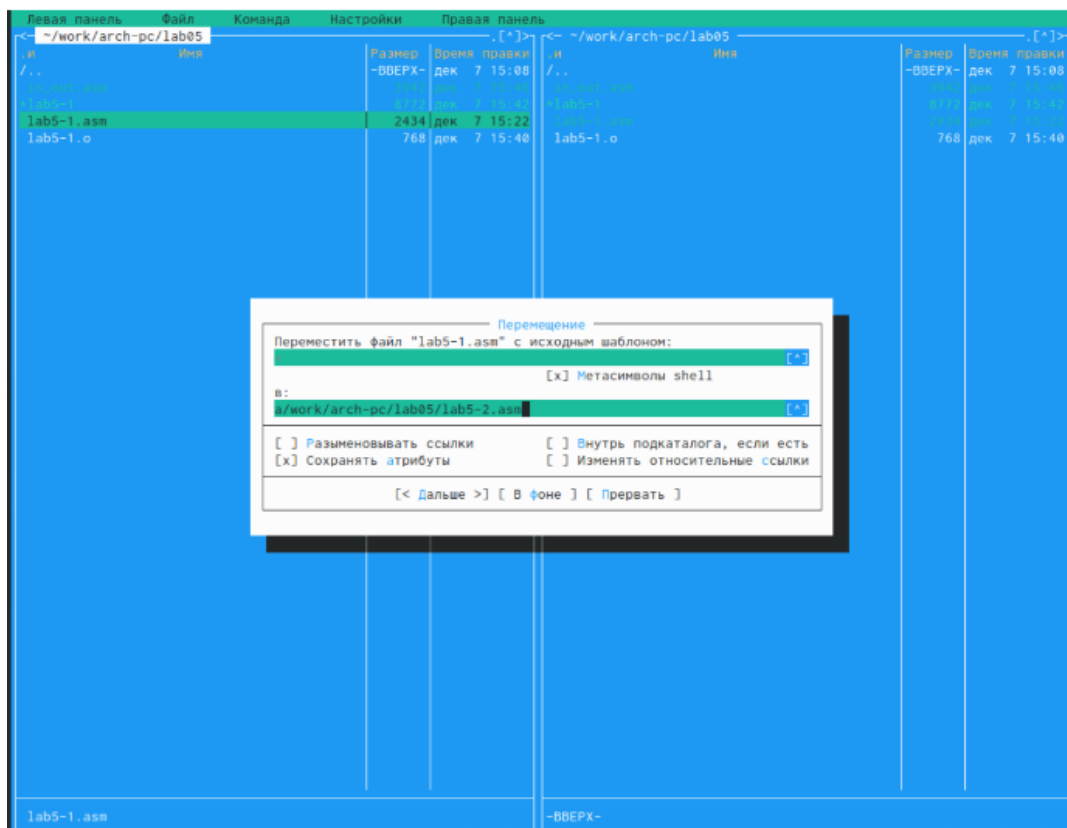


Рис. 3.8: Создание копии

9. Исправим текст программы в соответствии с листингом 6.2

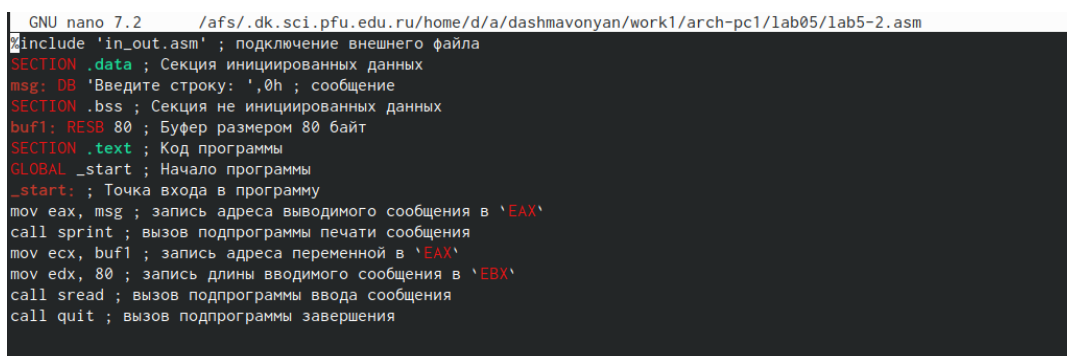


Рис. 3.9: Внесение изменений в файл

10. Оттранслируем текст программы lab5-2.asm в объектный файл и проверим его работоспособность

```

dashmavonyan@dk2n26 ~/work1/arch-pc1/lab05 $ nasm -f elf lab5-2.asm
dashmavonyan@dk2n26 ~/work1/arch-pc1/lab05 $ ld -m elf_i386 -o lab5-2 lab5-2.o
dashmavonyan@dk2n26 ~/work1/arch-pc1/lab05 $ ./lab5-2
Введите строку:
123

```

Рис. 3.10: Транслирование текста, проверка работоспособности

11. Исправьте текст программы, так чтобы она работала по следующему алгоритму:

- вывести приглашение типа “Введите строку:”
- ввести строку с клавиатуры;
- вывести введенную строку на экран.

```

GNU nano 7.2 /afs/.dk.sci.pfu.edu.ru/home/d/a/dashmavonyan/work1/arch-pc1/lab05/lab5-2.asm
#include 'in_out.asm' ; подключение внешнего файла
SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку: ',0h ; сообщение
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу

mov eax, msg ; запись адреса выводимого сообщения в 'EAX'
call sprintLF ; вызов подпрограммы печати сообщения

mov ecx, buf1 ; запись адреса переменной в 'EAX'
mov edx, 80 ; запись длины вводимого сообщения в 'EBX'
call sread ; вызов подпрограммы ввода сообщения

mov ecx, buf1 ; запись адреса переменной в 'EAX'
mov edx, 80 ; запись длины вводимого сообщения в 'EBX'
call sprintLF
call quit ; вызов подпрограммы завершения

```

Рис. 3.11: Проверка файла3

12. Оттранслируем текст программы в объектный файл. Внесём изменения в текст программы в файле lab5.asm

```
dashmavonyan@dk2n26 ~/work1/arch-pc1/lab05 $ nasm -f elf lab5-2.asm
dashmavonyan@dk2n26 ~/work1/arch-pc1/lab05 $ ld -m elf_i386 -o lab5-2 lab5-2.o
dashmavonyan@dk2n26 ~/work1/arch-pc1/lab05 $ ./lab5-2
Введите строку:
Шмавонян Диана
Шмавонян Диана
```

Рис. 3.12: Проверка работоспособности

4 Выводы

В ходе лабораторной работы мною были приобретены практические навыки работы в Midnight Commander, а также освоены инструкции языка ассемблера `mov` и `int`. Я научился работать с МС, и с его помощью работать с файлами (Создание, переименовывание, копирование, перемещение, удаление, и тд.)