

# **Лабораторная работа №6**

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

Лушин Артем Андреевич

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>3</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>4</b>
<b>3</b>	<b>Самостоятельная работа</b>	<b>10</b>
<b>4</b>	<b>Вывод</b>	<b>13</b>

# 1 Цель работы

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

## 2 Выполнение лабораторной работы

- 1) С помощью команды `mc` я открыл Midnight Commander и нашел папку с лабораторной 5

3) Через строку ввода я создал файл lab6-1.asm

.и	Имя	Размер	Время правки
/..		-ВВЕРХ-	ноя 13 02:24
	lab06-1.asm	0	ноя 13 02:26

Рис. 2.3: Создание файла lab6-1.asm

4) С помощью встроенного редактора я ввел текст программы и проверил его.

```
/home/aalushin/work/study/2022-~arch-pc/lab05/lab06/lab6-1.asm 285/285
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg

SECTION .bss

buf1: RESB 80
SECTION .text
GLOBAL _start
_start:

mov eax,4
mov ebx,1
mov ecx,msg
mov edx,msgLen
int 80h

mov eax, 3
mov ebx, 0
mov ecx, buf1
mov edx, 80
int 80h

mov eax,1
mov ebx,0
int 80h
```

Рис. 2.4: Текст в файле lab6-1.asm

5) Я оттранслировал текст в объектный файл. Выполнил компоновку и запустил файл программу.

```

[aalushin@fedora lab06]$ nasm -f elf lab6-1.asm
[aalushin@fedora lab06]$ ld -m elf_i386 -o lab6-1 lab6-1.o
[aalushin@fedora lab06]$ ls
lab6-1  lab6-1.asm  lab6-1.o
[aalushin@fedora lab06]$ ./lab6-1
Введите строку:
Лушин Артем Андреевич
[aalushin@fedora lab06]$

```

Рис. 2.5: Программа lab6-1

6) Я скачал файл in\_out.asm с ТУИС и перенес его в каталог lab06.

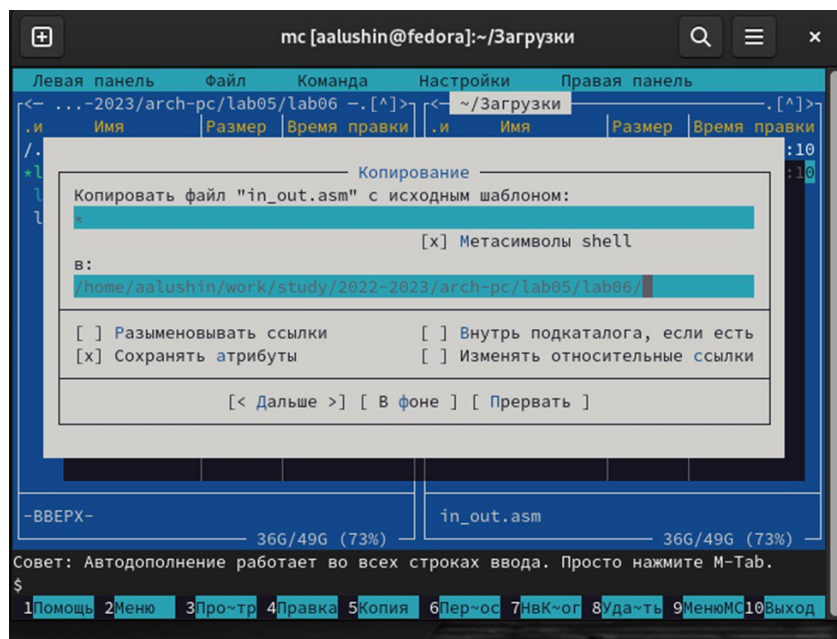


Рис. 2.6: Файл in\_out.asm

7) Я создал копию файла lab6-1.asm и назвал ее lab6-2.asm.

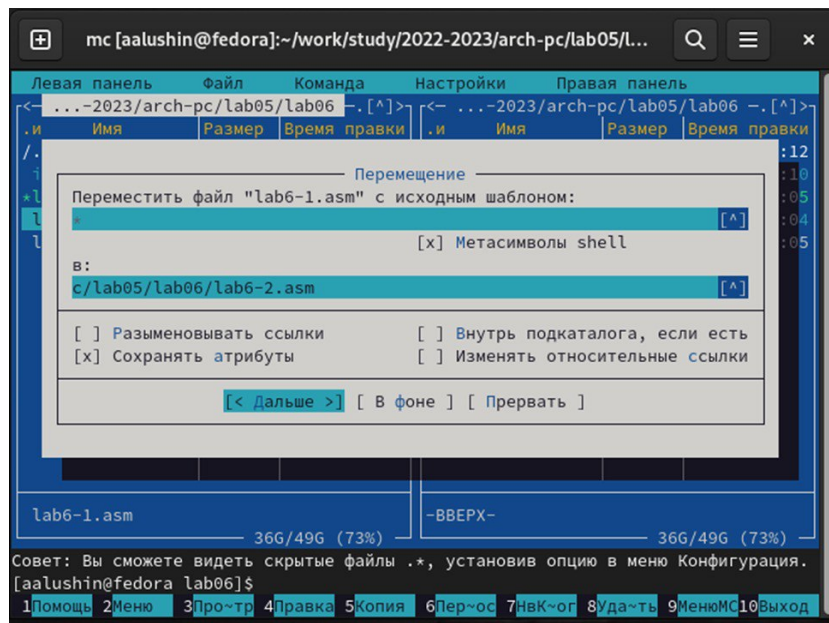


Рис. 2.7: Создание файла lab6-2.asm

- 8) Я исправил текст программы в файле lab6-2.asm с использованием программ из внешнего файла. Создал используемый файл и проверил его.

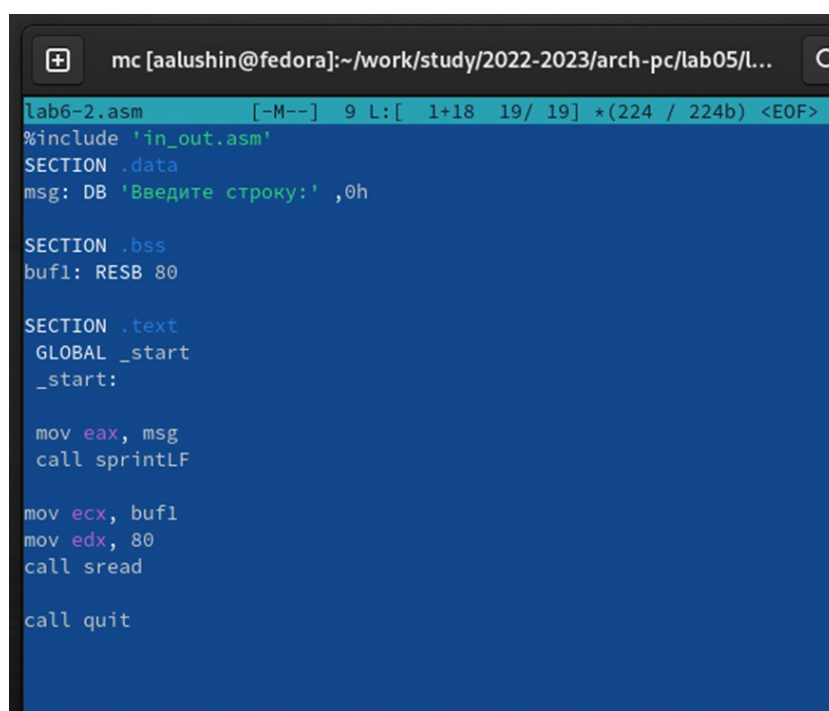


Рис. 2.8: Создание программы в файле lab6-2.asm

```

[aalushin@fedora lab06]$ nasm -f elf lab6-2.asm
[aalushin@fedora lab06]$ ld -m elf_i386 -o lab6-2 lab6-2.o
[aalushin@fedora lab06]$ ./lab6-2
Введите строку:
Пушин Артем
[aalushin@fedora lab06]$

```

Рис. 2.9: Проверка файла lab6-2

- 9) Я заменил `sprintf` на `sprint` в файле lab6-2. Проверил получившуюся программу. Разница между `sprintf` и `sprint` в том, что в первом случае мы вводим текст на другой строке, а во втором, на той же, где и текст запроса.

```

lab6-2.asm [----] 12 1: [ 1+12 13/
%include 'in_out.asm'
SECTION .data
msg: DB 'Введите строку:' ,0h

SECTION .bss
buf1: RESB 80

SECTION .text
GLOBAL _start
_start:

    mov eax, msg
    call sprint

    mov ecx, buf1
    mov edx, 80
    call sread

    call quit

```

Рис. 2.10: Программа lab6-2 с изменениями



```
[aalushin@fedora lab06]$ nasm -f elf lab6-2.asm  
[aalushin@fedora lab06]$ ld -m elf_i386 -o lab6-2 lab6-2.o  
[aalushin@fedora lab06]$ ./lab6-2  
Введите строку:Лушин Артем  
[aalushin@fedora lab06]$
```

Рис. 2.11: Проверка программы lab6-2 с изменениями

### 3 Самостоятельная работа

- 1) Я создал копию файла lab6-1.asm и назвал ее lab6-3.asm. Ввел текст чтобы программа выводила нужный результат.

```
/home/aalushin/work/study/2022-2023/arch-pc/lab05/lab06/lab6-3.asm
SECTION .data
msg: DB 'Введите строку:',10

msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80

SECTION .text
GLOBAL _start
_start:

mov eax,4
mov ebx,1
mov ecx,msg
mov edx,msgLen
int 80h

mov eax, 3
mov ebx, 0
mov ecx, buf1
mov edx, 80
int 80h

mov eax, 4
mov ebx,1
mov ecx, buf1
int 80h

mov eax,1
mov ebx,0
int 80h
```

Рис. 3.1: Создание файла lab6-3 и действия с ним

- 2) Я проверил работу файла lab6-3.

```

[aalushin@fedora lab06]$ nasm -f elf lab6-3.asm
[aalushin@fedora lab06]$ ld -m elf_i386 -o lab6-3 lab6-3.o
[aalushin@fedora lab06]$ ./lab6-3
Введите строку:
Лушин Атем
Лушин Атем
[aalushin@fedora lab06]$ ./lab6-3
Введите строку:
ЛУШИН Артем
ЛУШИН Артем

```

Рис. 3.2: Проверка файла lab6-3

- 3) Я создал копию файла lab6-2.asm и назвал ее lab6-4.asm. Ввел текст чтобы программа выводила нужный результат.

```

%include 'in_out.asm'

SECTION .data
msg: DB 'Введите строку:',10

SECTION .bss
buf1: RESB 80

SECTION .text
GLOBAL _start
_start:

    mov eax, msg
    call sprint

    mov ecx, buf1
    mov edx, 80
    call sread
    mov eax, buf1
    call sprint

    call quit

```

Рис. 3.3: Создание файла lab6-4.asm

- 4) Я проверил файл lab6-4.

```

[aalushin@fedora lab06]$ nasm -f elf lab6-4.asm
[aalushin@fedora lab06]$ ld -m elf_i386 -o lab6-4 lab6-4.o
[aalushin@fedora lab06]$ ./lab6-4
Введите строку:
Лушин Ароем
Лушин Ароем
[aalushin@fedora lab06]$ ./lab6-4
Введите строку:
Лушин Артем
Лушин Артем

```

Рис. 3.4: Проверка файла lab6-4

5) Я загрузил результаты работы на GitHub.

```

create mode 190644 labs/lab05/report/image/photo1668014608.jpeg
create mode 190644 labs/lab05/report/image/photo1668014626.jpeg
create mode 190644 labs/lab05/report/image/photo1668014637.jpeg
create mode 190644 labs/lab05/report/image/photo1668014646.jpeg
create mode 190644 labs/lab05/report/lab05.tar
create mode 190644 labs/lab05/report/report.docx
create mode 190644 labs/lab05/report/report.pdf
create mode 190644 labs/lab05/report/selnolig.sty
create mode 190644 labs/lab06/lab6-1.asm
create mode 190644 labs/lab06/lab6-2.asm
create mode 190644 labs/lab06/lab6-3.asm
create mode 190644 labs/lab06/lab6-4.asm
create mode 190644 labs/lab06/report/image/photo1668813343.jpeg
create mode 190644 labs/lab06/report/image/photo1668813353.jpeg
create mode 190644 labs/lab06/report/image/photo1668813354.jpeg
create mode 190644 labs/lab06/report/image/photo1668813366.jpeg
create mode 190644 labs/lab06/report/image/photo1668813370.jpeg
create mode 190644 labs/lab06/report/image/photo1668813375.jpeg
create mode 190644 labs/lab06/report/image/photo1668813379.jpeg
create mode 190644 labs/lab06/report/image/photo1668813390.jpeg
create mode 190644 labs/lab06/report/image/photo1668813395.jpeg
create mode 190644 labs/lab06/report/image/photo1668813403.jpeg
create mode 190644 labs/lab06/report/image/photo1668813408.jpeg
create mode 190644 labs/lab06/report/image/photo1668813424.jpeg
create mode 190644 labs/lab06/report/image/photo1668813431.jpeg
create mode 190644 labs/lab06/report/image/photo1668813508.jpeg
create mode 190644 labs/lab06/report/image/photo1668813520.jpeg
create mode 190644 labs/lab06/report/report.docx
create mode 190644 labs/lab06/report/report.pdf
create mode 190644 labs/lab06/report/selnolig.sty
aalushin@10 study_2022-2023_arh-pc]$ git push
it: «push» не является командой git. Смотрите «git --help».

аные похожие команды:
push
aalushin@10 study_2022-2023_arh-pc]$ git push
ерчисление объектов: 55, готово.
дсчет объектов: 100% (54/54), готово.
ри сжатии изменений используется до 2 потоков
жатие объектов: 100% (47/47), готово.
пись объектов: 100% (47/47), 5.57 МБ | 1.11 МБ/с, готово.
сею 47 (изменений 5), повторно использовано 0 (изменений 0), повторно использовано пакетов 0
emote: Resolving deltas: 100% (5/5), completed with 3 local objects.
o github.com:SkLjT/study_2022-2023_arh-pc.git
722831b..7df4b7d master -> master
aalushin@10 study_2022-2023_arh-pc]$

```

Рис. 3.5: Загрузка на GitHub

## 4 Вывод

Я приобрел практические навыки работы с Midnight Commander и освоил инструкцию языка ассемблера `mov` и `int`.