

Отчёт по лабораторной работе 6

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

Артем Абрикосов НПИбд-01-22

Содержание

1	Цель работы:.....	1
2	Порядок выполнения лабораторной работы:.....	1
3	Порядок выполнения самостоятельной работы:.....	8
4	Вывод:.....	10

1 Цель работы:

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

2 Порядок выполнения лабораторной работы:

Откроем Midnight Commander и перейдем в каталог ~/work/arch-pc созданный при выполнении лабораторной работы №5 (рис.1).

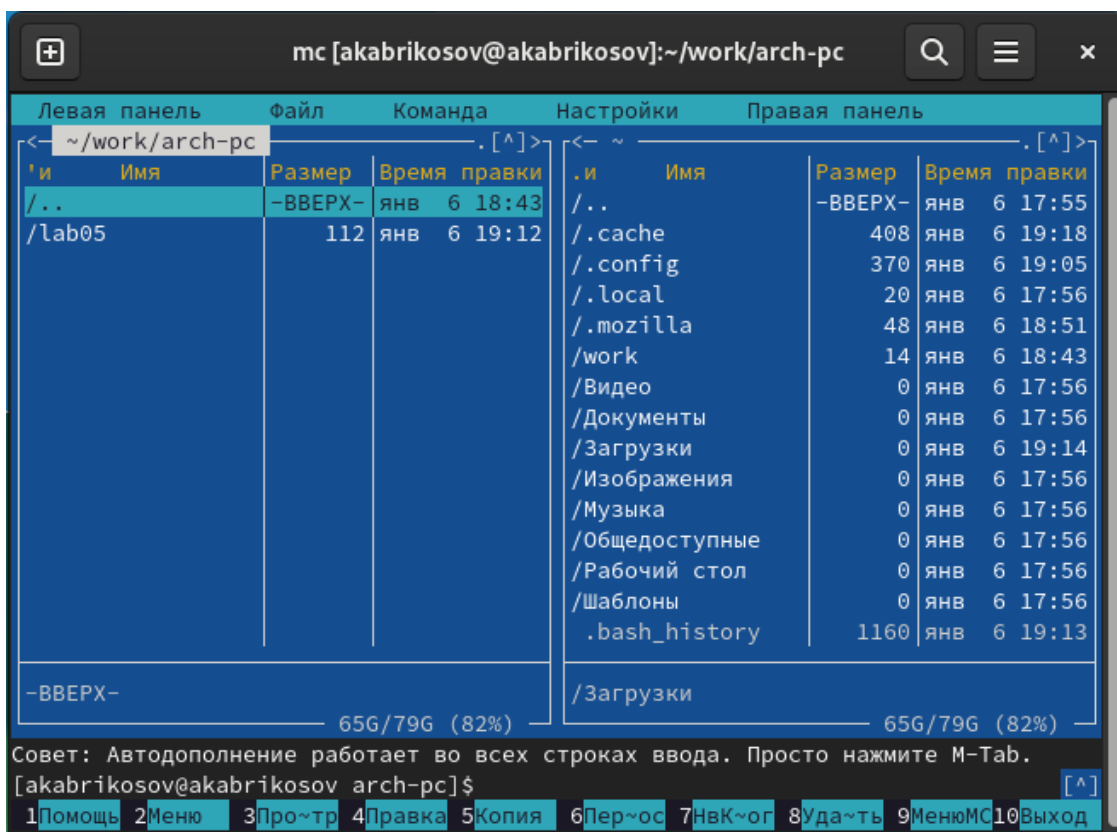


Рис. 1. МС, каталог ~/work/arch-pc

Затем создадим папку lab06 и создадим в ней файл lab6-1.asm (рис. 2).

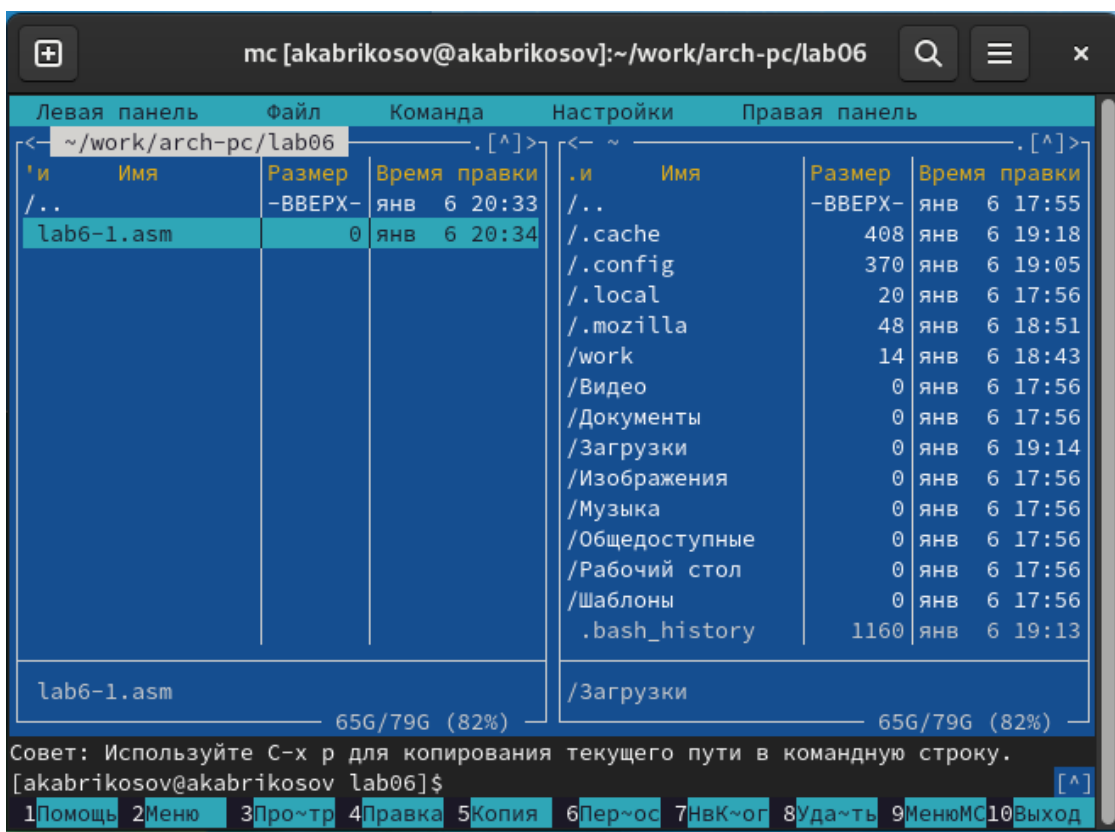


Рис. 2. Создание папки lab06 и файла lab6-1.asm

Потом откроем файл lab6-1.asm и введем следующий текст программы (рис. 3.)

```
mc [akabrikosov@aka
lab6-1.asm [----] 0 L:[ 1-
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. 3. Текст программы lab6-1.asm

Затем оттранслируем текст программы lab6-1.asm в объектный файл. Выполним компоновку объектного файла и запустим получившийся исполняемый файл. Затем на запрос программы введем свои ФИО (рис. 4).

```
[akabrikosov@akabrikosov lab06]$ nasm -f elf lab6-1.asm
[akabrikosov@akabrikosov lab06]$ ld -m elf_i386 lab6-1.o -o lab6-1
[akabrikosov@akabrikosov lab06]$ ls
lab6-1 lab6-1.asm lab6-1.o
[akabrikosov@akabrikosov lab06]$ ./lab6-1
Введите строку:
Абрикосов Артем Камович
[akabrikosov@akabrikosov lab06]$
```

Рис. 4. Программа lab6-1

Скачаем файл `in_out.asm` со страницы курса в ТУИС, чтобы в дальнейшем упростить нашу жизнь, и переместим его в каталог с файлом `lab6-1.asm` (рис. 5).

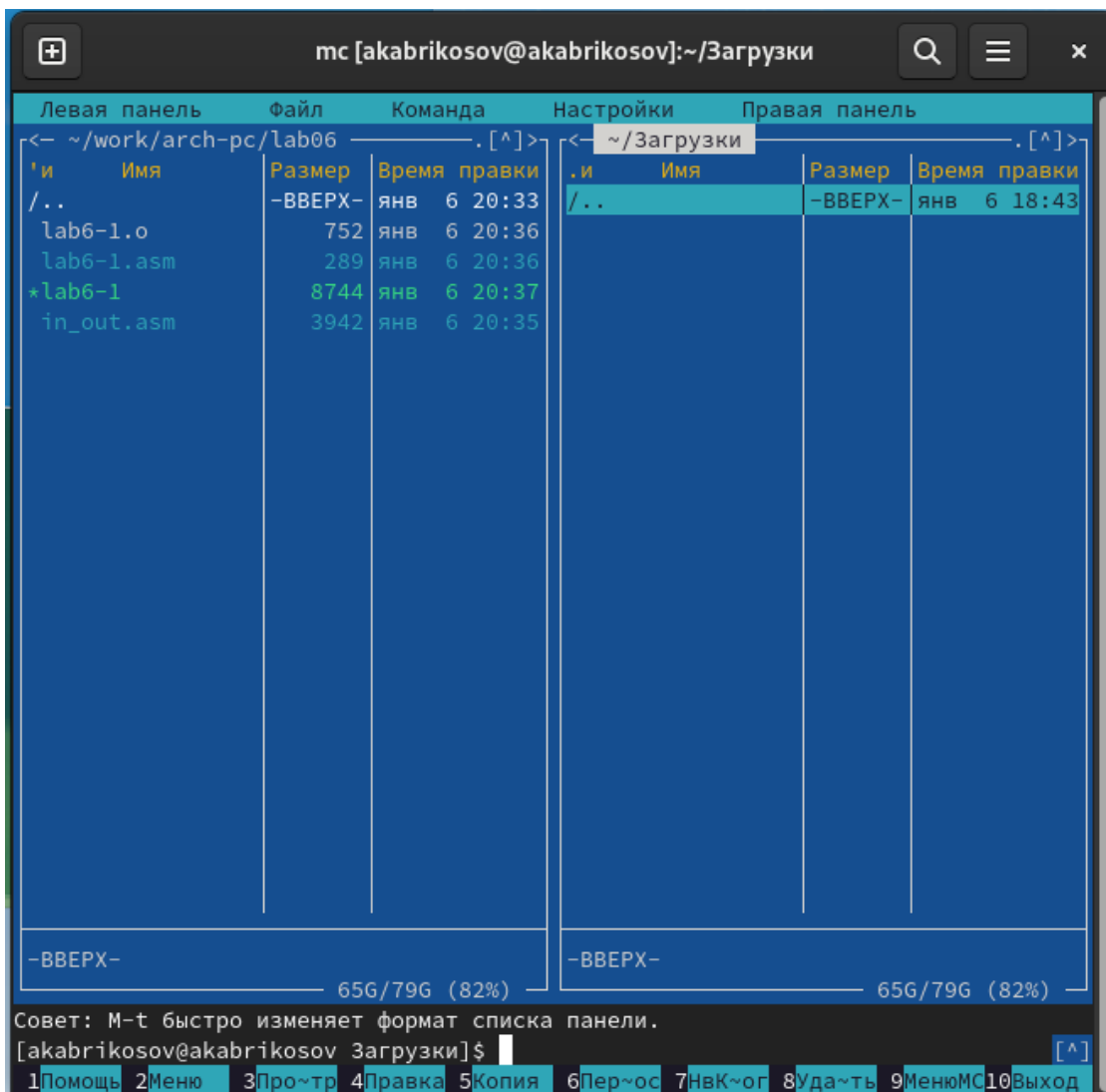


Рис. 5. Копирование файла `in_out.asm`

Создадим копию файла `lab6-1.asm` с именем `lab6-2.asm` (рис. 6). Затем исправим текст программы в файле `lab6-2.asm` с использованием подпрограмм из внешнего файла `in_out.asm` (рис. 7). Создадим исполняемый файл и проверим его работу (рис. 8)

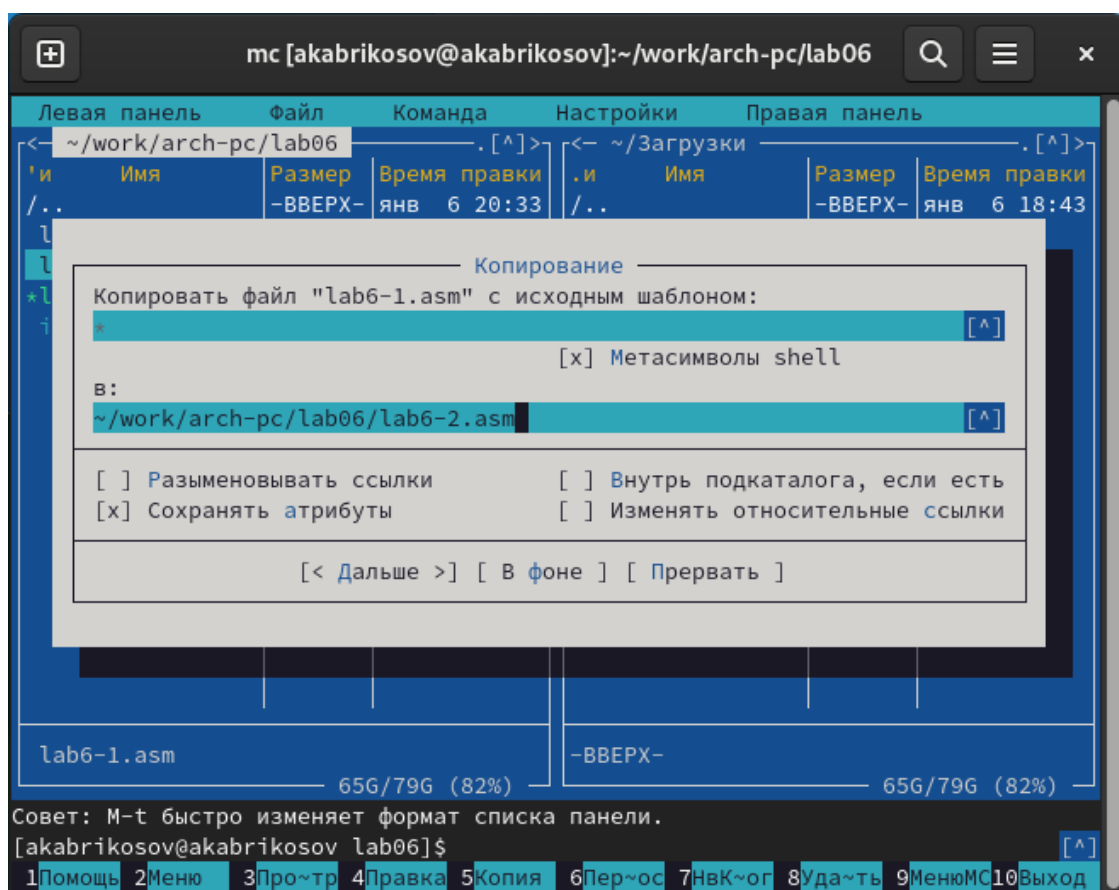
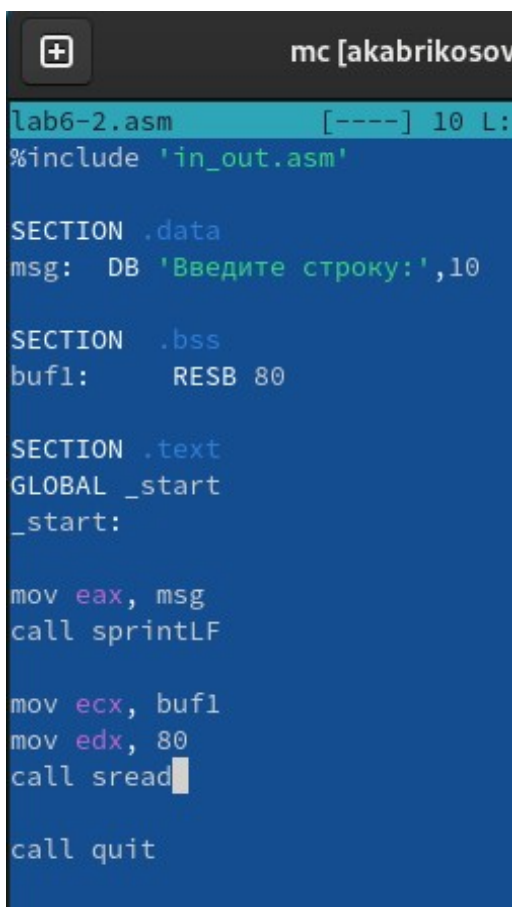


Рис. 6. Копирование файла lab6-1.asm

A screenshot of a text editor window titled 'mc [akabrikosov]'. The editor displays the assembly code for 'lab6-2.asm'. The code includes a header file 'in_out.asm', defines a data section with a message 'Введите строку:', a bss section for a buffer 'buf1' of size 80, and a text section with the entry point '_start'. The main logic involves moving the message to 'eax', calling 'sprintLF', moving the buffer address to 'ecx', setting 'edx' to 80, calling 'sread', and finally calling 'quit'.

```
lab6-2.asm [-----] 10 L:
#include 'in_out.asm'

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

SECTION .bss
buf1: RESB 80

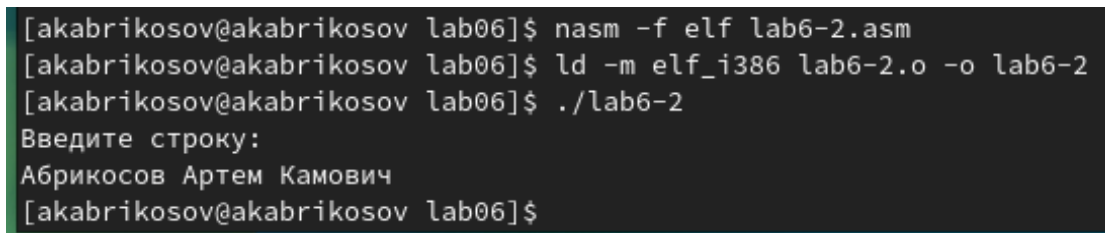
SECTION .text
GLOBAL _start
_start:

mov eax, msg
call sprintLF

mov ecx, buf1
mov edx, 80
call sread

call quit
```

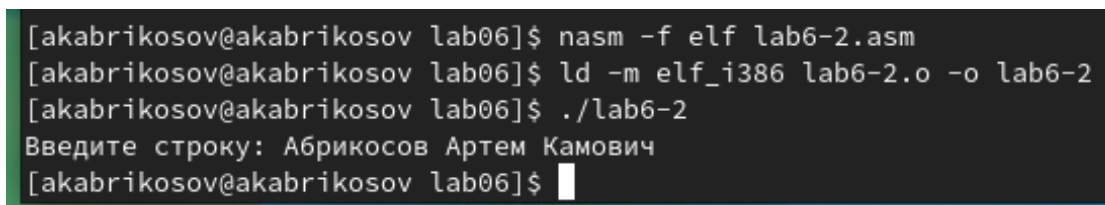
Рис. 7. Исправление текста программы

A screenshot of a terminal window showing the compilation and execution of the program. The commands used are 'nasm -f elf lab6-2.asm', 'ld -m elf_i386 lab6-2.o -o lab6-2', and './lab6-2'. The output shows the prompt 'Введите строку:' followed by the input 'Абрикосов Артем Камович' and a new line.

```
[akabrikosov@akabrikosov lab06]$ nasm -f elf lab6-2.asm
[akabrikosov@akabrikosov lab06]$ ld -m elf_i386 lab6-2.o -o lab6-2
[akabrikosov@akabrikosov lab06]$ ./lab6-2
Введите строку:
Абрикосов Артем Камович
[akabrikosov@akabrikosov lab06]$
```

Рис. 8. Результат работы программы lab6-2 (1)

Затем в этом же файле заменим подпрограмму `sprintLF` на `sprint` и затем запустим (рис. 9).

A screenshot of a terminal window showing the compilation and execution of the program with the 'sprint' function. The commands are the same as in the previous screenshot. The output shows the prompt 'Введите строку:' followed by the input 'Абрикосов Артем Камович' on the same line.


```
[akabrikosov@akabrikosov lab06]$ nasm -f elf lab6-2.asm
[akabrikosov@akabrikosov lab06]$ ld -m elf_i386 lab6-2.o -o lab6-2
[akabrikosov@akabrikosov lab06]$ ./lab6-2
Введите строку: Абрикосов Артем Камович
[akabrikosov@akabrikosov lab06]$
```

Рис. 9. Запуск программы lab6-2 (2)

И видим, что по сравнению со `sprintLF` с подпрограммой `sprint` нет переноса строки после вывода строки программы.

3 Порядок выполнения самостоятельной работы:

Создадим копию файла lab6-1.asm и назовем его lab6-1_1.asm. Внесем изменения в программу, так чтобы она после ввода выводила напечатанную строку (рис. 10). И запустим её (рис. 11).



```
mc [akabrikosov@
lab6-1_1.asm [----] 7 L: [
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, 0
mov ecx, buf1
mov edx, 80
int 80h

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

Рис. 10. Изменения в программе lab6-1_1.asm


```

[akabrikosov@akabrikosov lab06]$ nasm -f elf lab6-1_1.asm
[akabrikosov@akabrikosov lab06]$ ld -m elf_i386 lab6-1_1.o -o lab6-1_1
[akabrikosov@akabrikosov lab06]$ ./lab6-1_1
Введите строку:
Абрикосов Артем Камович
Абрикосов Артем Камович
[akabrikosov@akabrikosov lab06]$

```

Рис. 11. Работа программы lab6-1_1

Затем сделаем этот процесс со второй программой (рис. 12-14).

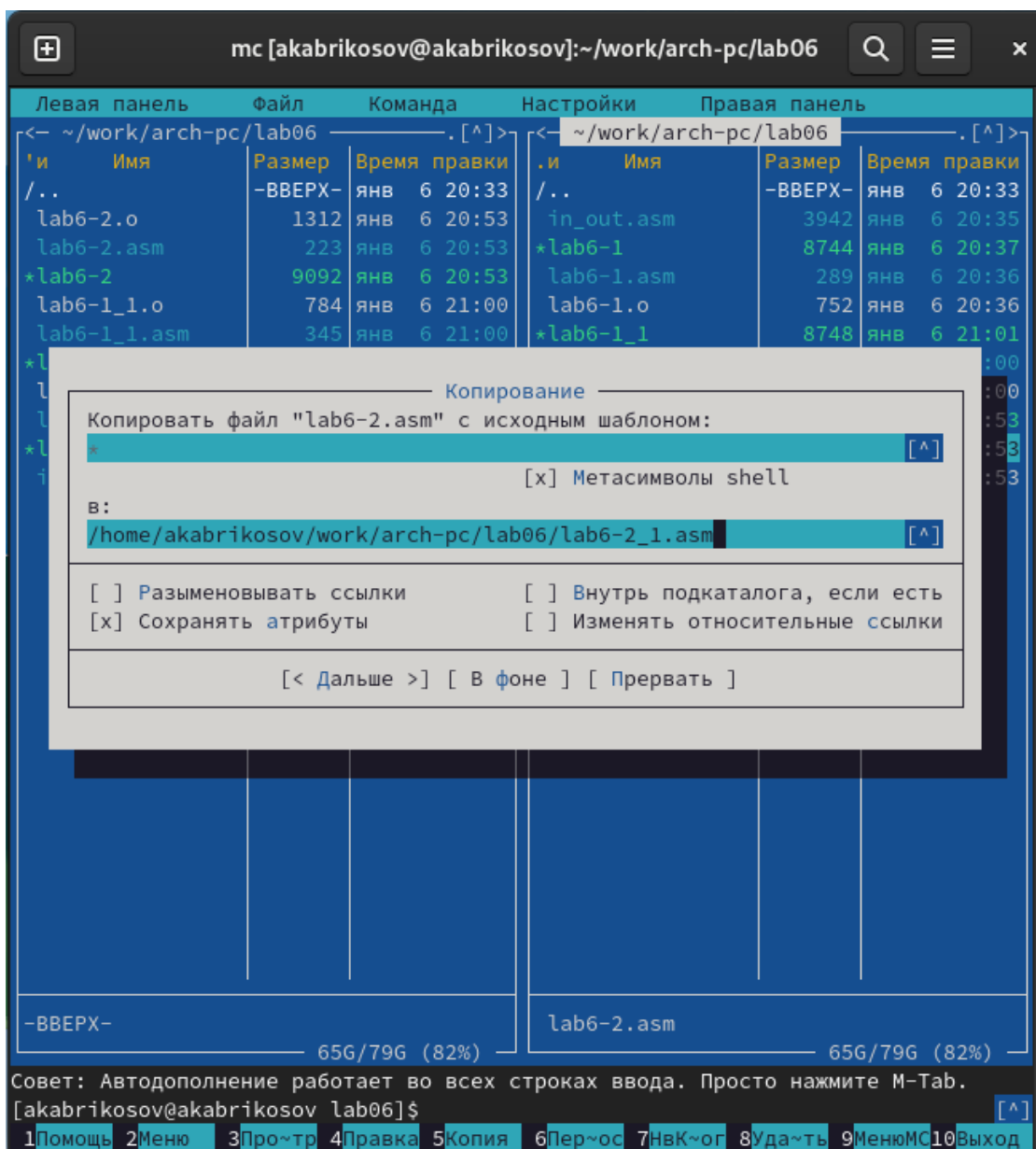
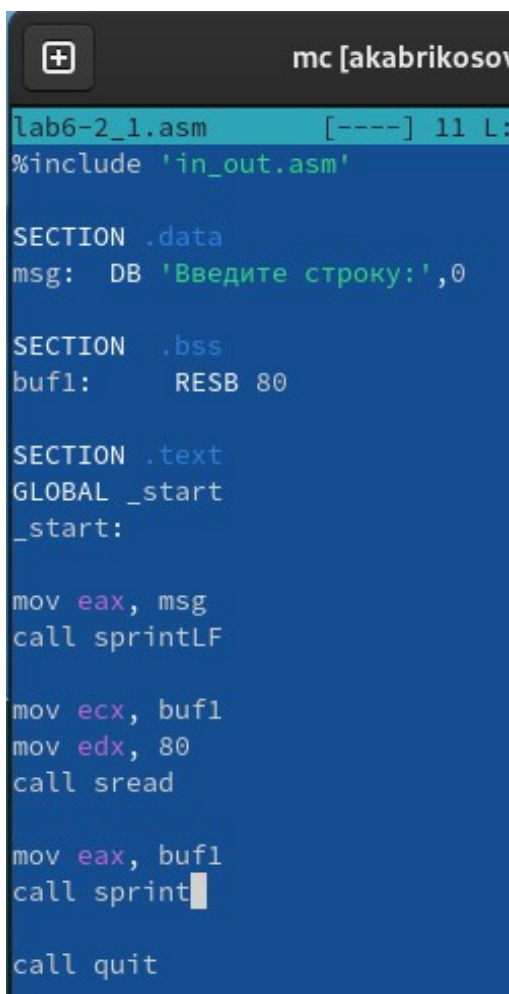


Рис. 12. Копия файла lab6-2.asm в файл lab6-2_1.asm

A screenshot of a text editor window titled 'mc [akabrikosov]'. The editor displays assembly code for 'lab6-2_1.asm'. The code includes an include directive for 'in_out.asm', followed by three sections: .data, .bss, and .text. The .data section defines a message string 'Введите строку:'. The .bss section reserves 80 bytes for a buffer. The .text section contains the main logic: it starts at '_start', moves the address of 'msg' to 'eax', calls 'sprintf', moves the address of 'buf1' to 'ecx', moves '80' to 'edx', calls 'sread', moves the address of 'buf1' to 'eax', calls 'sprintf', and finally calls 'quit'.

```
lab6-2_1.asm [----] 11 L:
#include 'in_out.asm'

SECTION .data
msg: DB 'Введите строку:',0

SECTION .bss
buf1: RESB 80

SECTION .text
GLOBAL _start
_start:

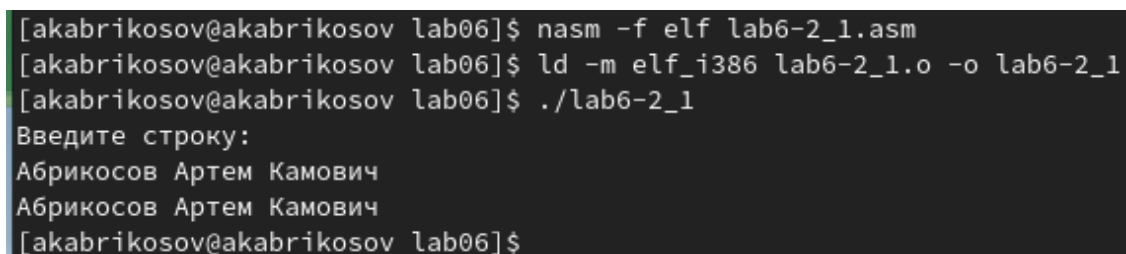
mov eax, msg
call sprintf

mov ecx, buf1
mov edx, 80
call sread

mov eax, buf1
call sprintf

call quit
```

Рис. 13. Изменения в программе lab6-2_1.asm

A screenshot of a terminal window showing the compilation and execution of the assembly program. The user runs 'nasm -f elf lab6-2_1.asm', then 'ld -m elf_i386 lab6-2_1.o -o lab6-2_1', and finally './lab6-2_1'. The program prompts 'Введите строку:' and the user enters 'Абрикосов Артем Камович' twice.

```
[akabrikosov@akabrikosov lab06]$ nasm -f elf lab6-2_1.asm
[akabrikosov@akabrikosov lab06]$ ld -m elf_i386 lab6-2_1.o -o lab6-2_1
[akabrikosov@akabrikosov lab06]$ ./lab6-2_1
Введите строку:
Абрикосов Артем Камович
Абрикосов Артем Камович
[akabrikosov@akabrikosov lab06]$
```

Рис. 14. Работа программы lab6-2_1

4 Вывод:

Во время выполнения лабораторной работы были приобретены практические навыки работы в Midnight Commander, а также освоены инструкции языка ассемблера mov и int.