

Отчёт по лабораторной работе 5

Дисциплина архитектура компьютера

Неустроева Ирина

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Midnight Commander.	6
2.2	Подключение внешнего файла in_out.asm	10
2.3	Задания для самостоятельной работы	14
3	Выводы	18

Список иллюстраций

2.1	Midnight Commander	6
2.2	Создание каталога	7
2.3	Создание файла lab05-1.asm	7
2.4	Программа в файле lab05-1.asm	8
2.5	Просмотр файла lab05-1.asm	9
2.6	Запуск программы lab05-1.asm	9
2.7	Копирование файла	10
2.8	Копирование файла	11
2.9	Программа в файле lab05-2.asm	12
2.10	Запуск программы lab05-2.asm	12
2.11	Программа в файле lab05-2.asm	13
2.12	Запуск программы lab05-2.asm	13
2.13	Программа в файле lab05-3.asm	14
2.14	Запуск программы lab05-3.asm	15
2.15	Программа в файле lab05-4.asm	16
2.16	Запуск программы lab05-4.asm	16

Список таблиц

1 Цель работы

Целью работы является приобретение практических навыков работы в Midnight Commander. Освоение инструкций языка ассемблера `mov` и `int`.

2 Выполнение лабораторной работы

2.1 Midnight Commander.

1 Открыла Midnight Commander. (рис. [2.1])

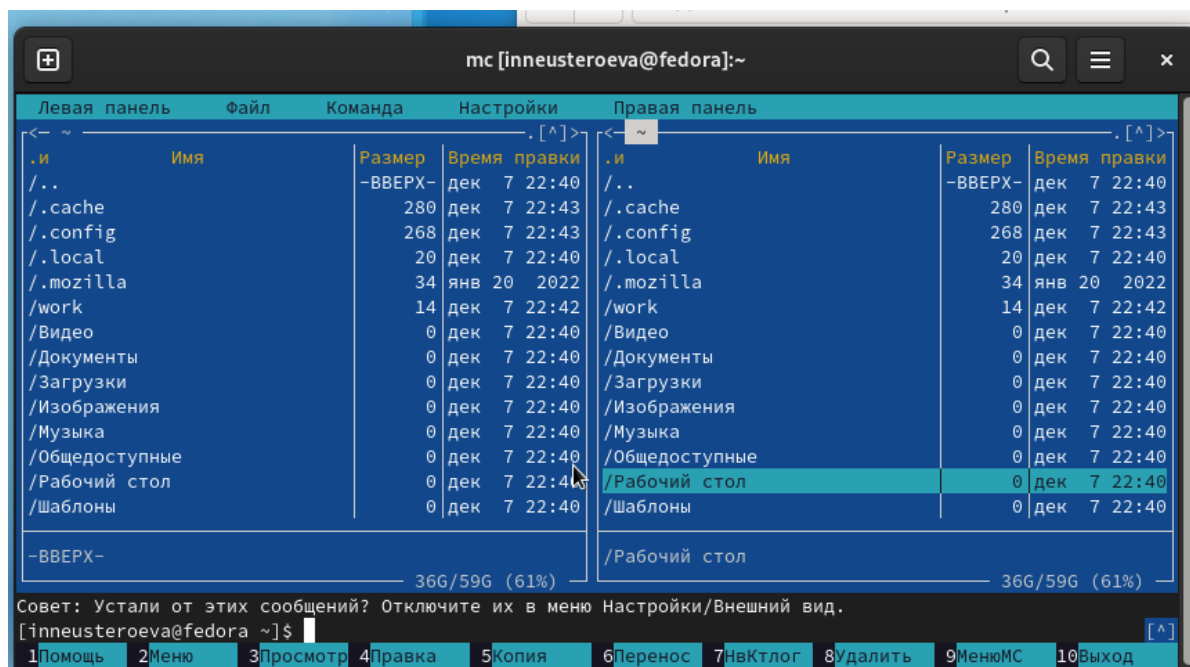


Рис. 2.1: Midnight Commander

2 Перешла в каталог ~/work/arch-рс. Создала каталог lab05 (рис. [2.2])

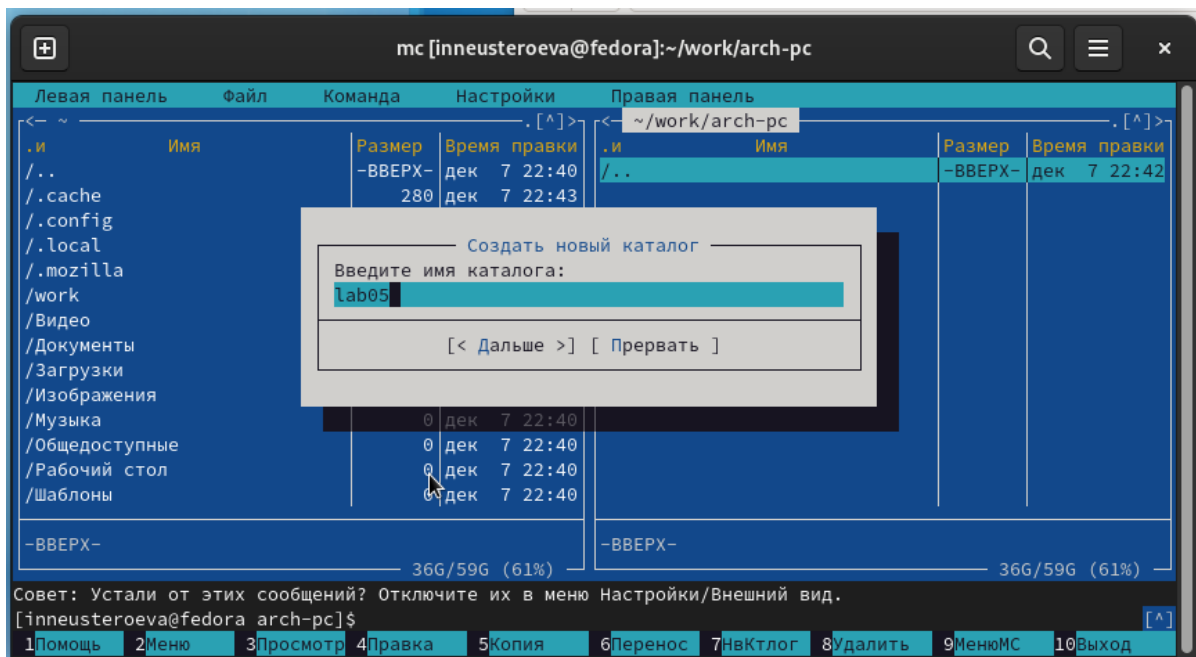


Рис. 2.2: Создание каталога

3 Создала файл lab05-1.asm (рис. [2.3])

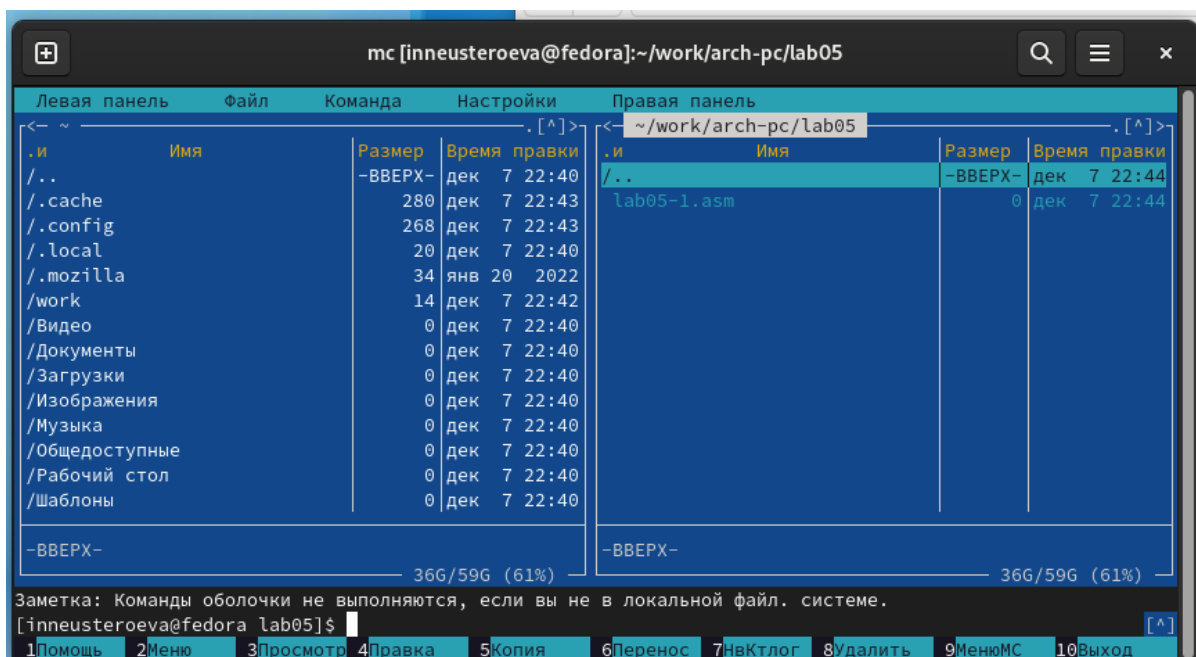
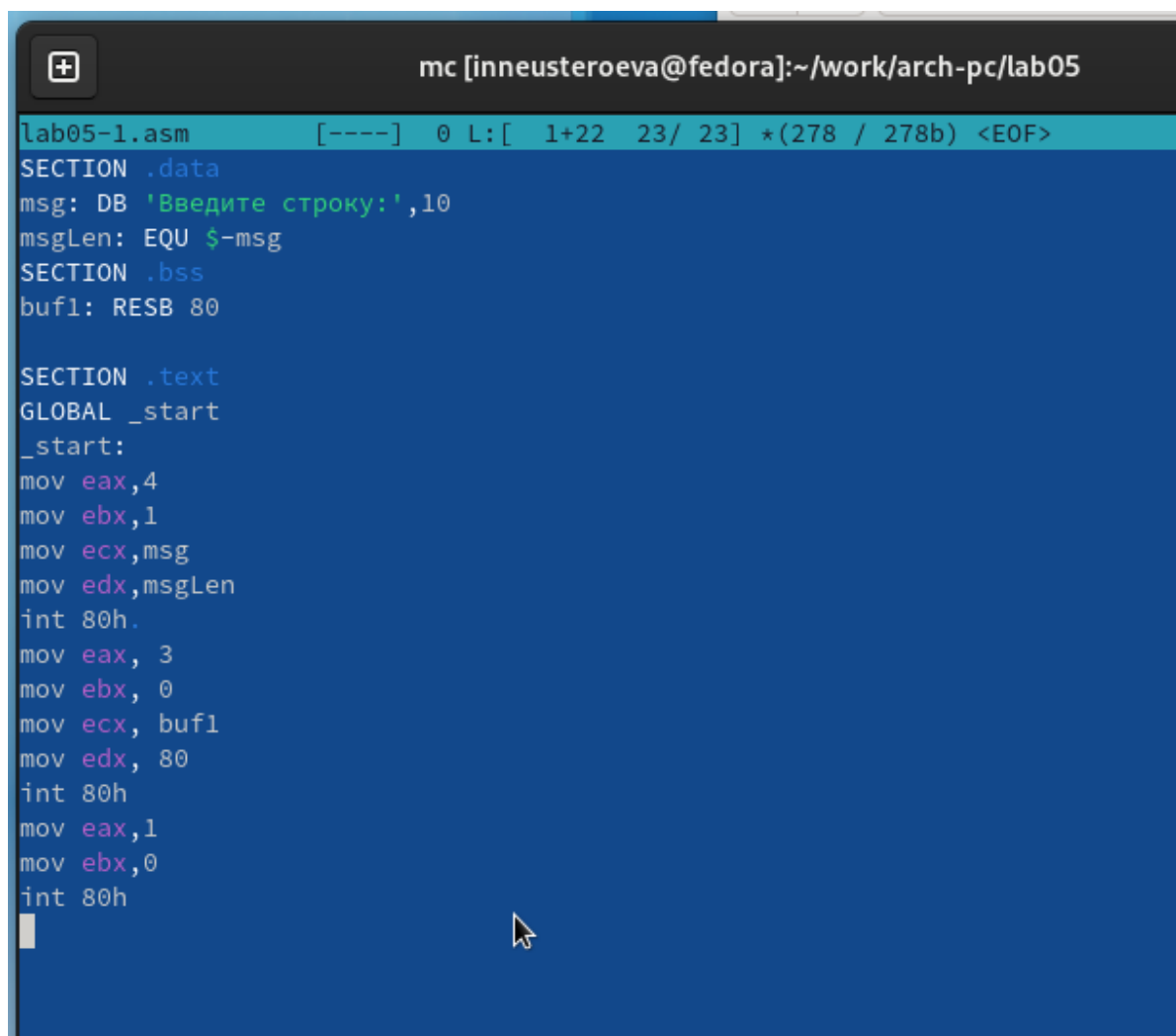


Рис. 2.3: Создание файла lab05-1.asm

4 Открыла файл на редактирование. Написал код. (рис. [2.4])

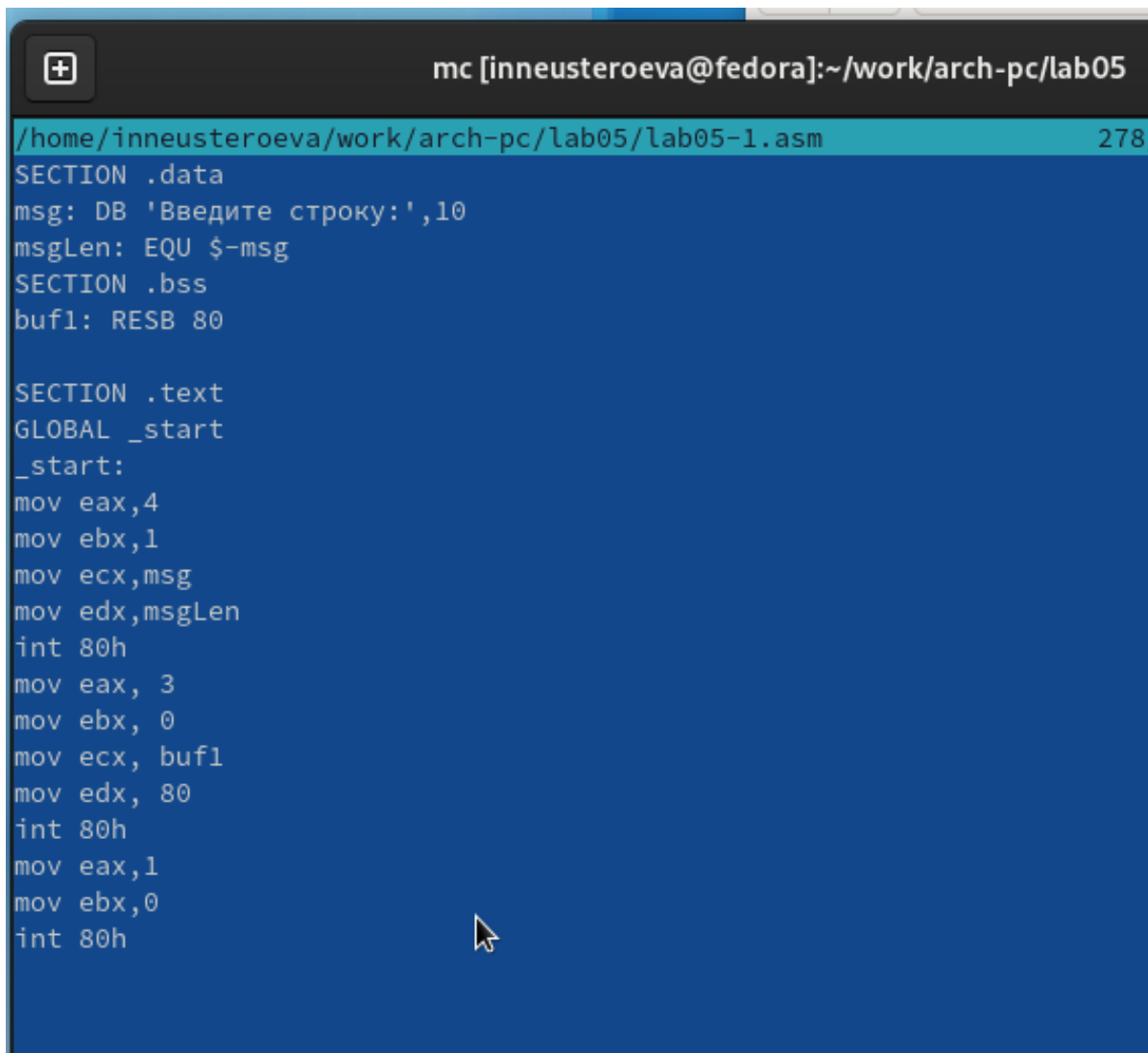


```
mc [inneusteroeva@fedora]:~/work/arch-pc/lab05
lab05-1.asm [----] 0 L: [ 1+22 23/ 23] *(278 / 278b) <EOF>
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

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: Программа в файле lab05-1.asm

5 Открыла файл для просмотра и проверила, что он содержит написанный код. (рис. [2.5])

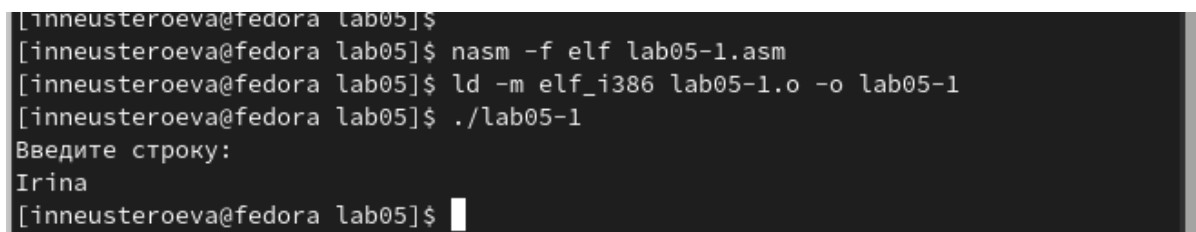


```
mc [inneusteroeva@fedora]:~/work/arch-pc/lab05
/home/inneusteroeva/work/arch-pc/lab05/lab05-1.asm 278
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

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. 2.5: Просмотр файла lab05-1.asm

6 Получила исполняемый файл программы и проверила его работу.(рис. [2.6])



```
[inneusteroeva@fedora lab05]$
[inneusteroeva@fedora lab05]$ nasm -f elf lab05-1.asm
[inneusteroeva@fedora lab05]$ ld -m elf_i386 lab05-1.o -o lab05-1
[inneusteroeva@fedora lab05]$ ./lab05-1
Введите строку:
Irina
[inneusteroeva@fedora lab05]$
```

Рис. 2.6: Запуск программы lab05-1.asm

2.2 Подключение внешнего файла in_out.asm

1 Скачала файл in_out.asm. Добавила файл in_out.asm в рабочий каталог.(рис. [2.7])

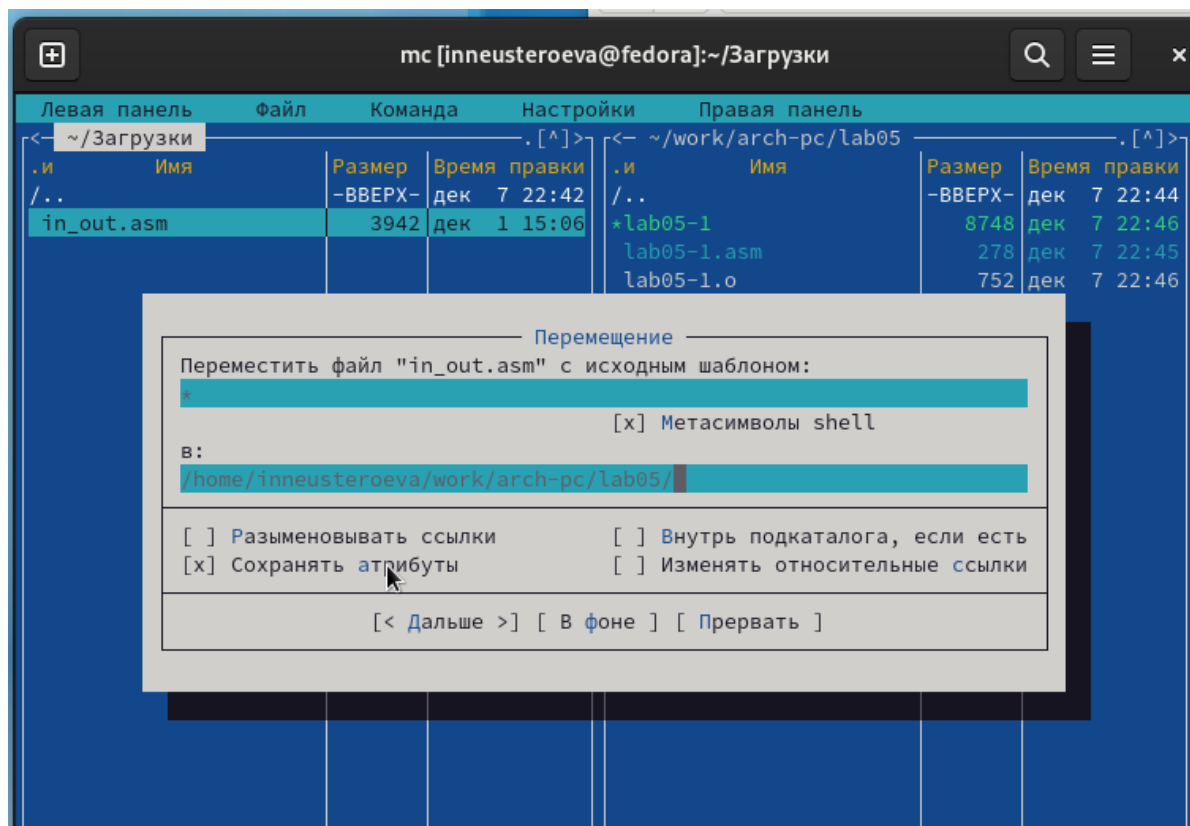


Рис. 2.7: Копирование файла

2 Скопировала lab05-1.asm в lab05-2.asm. (рис. [2.8])

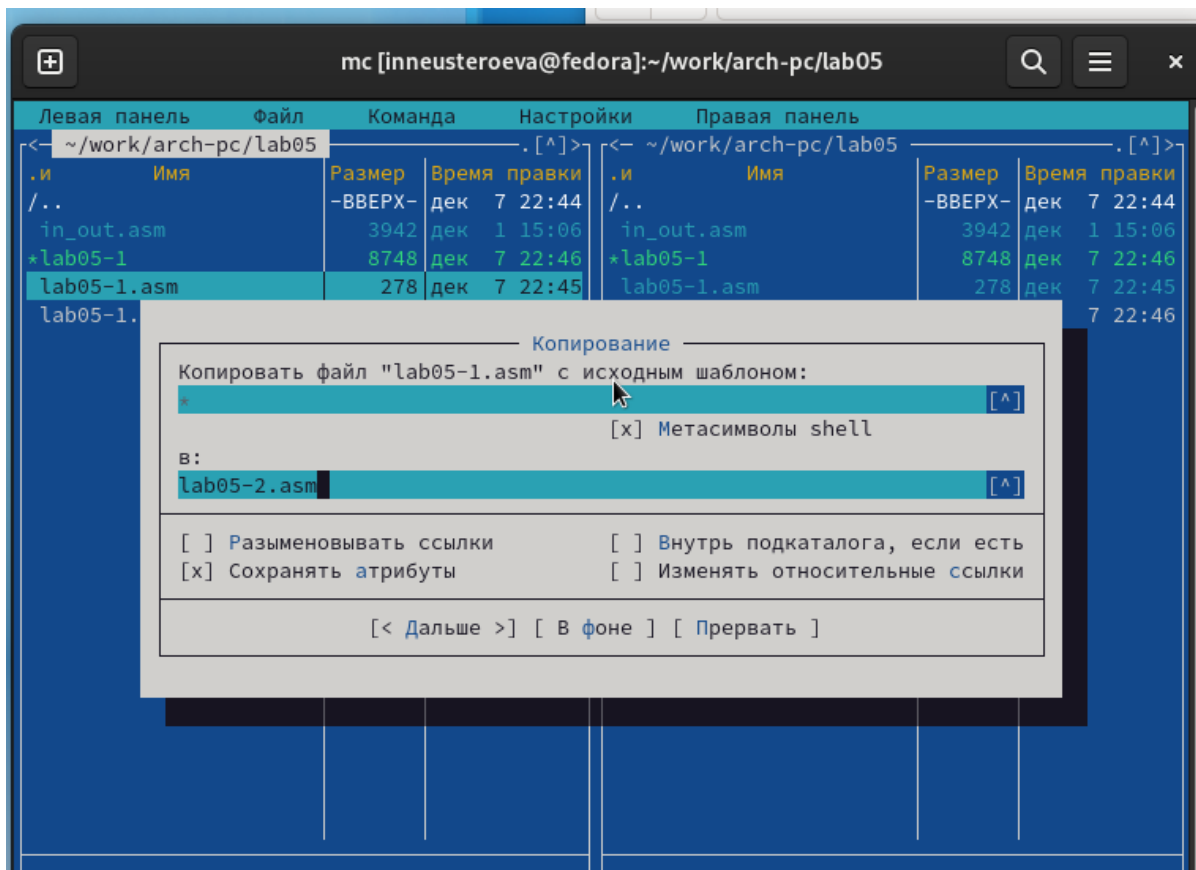
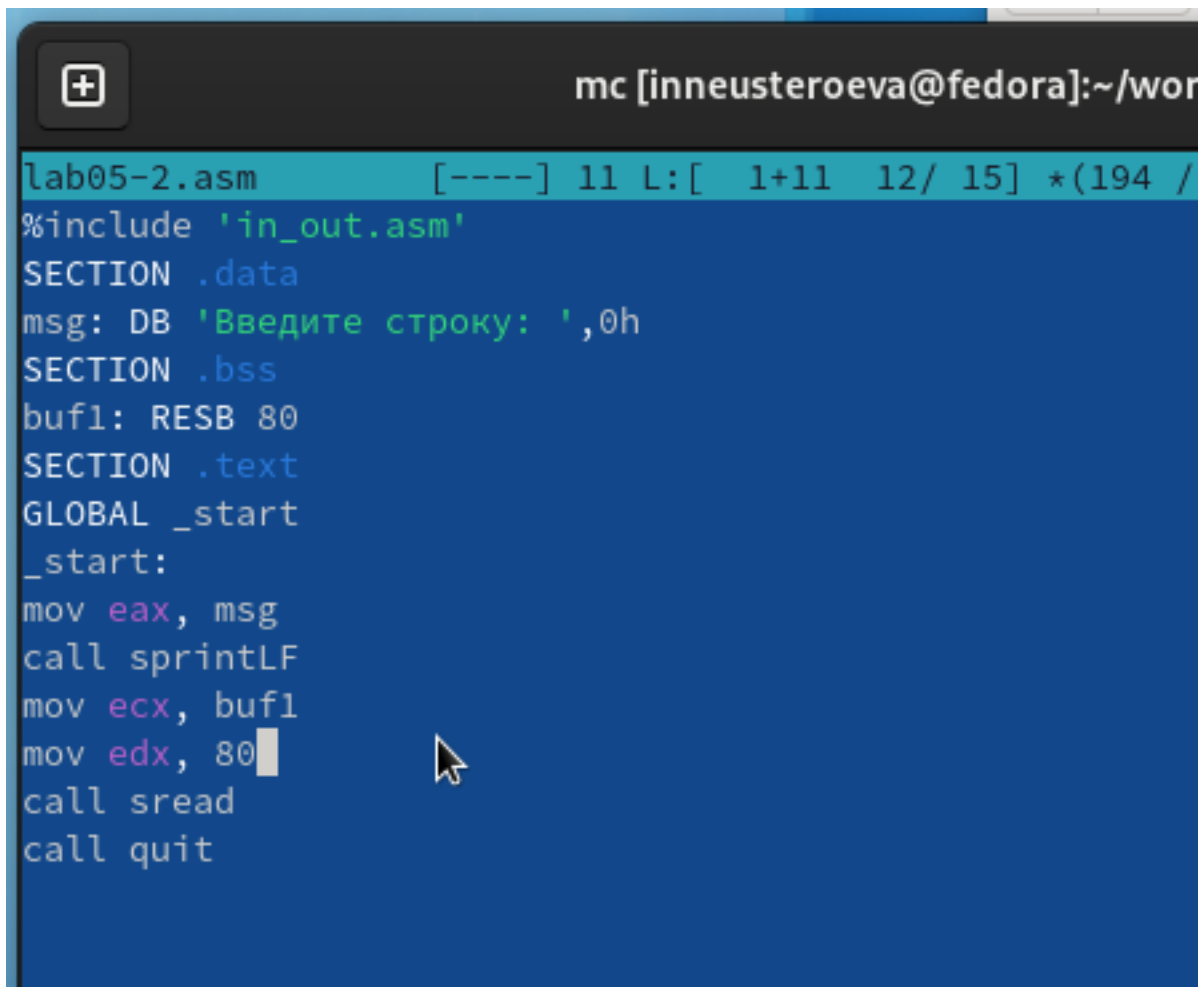


Рис. 2.8: Копирование файла

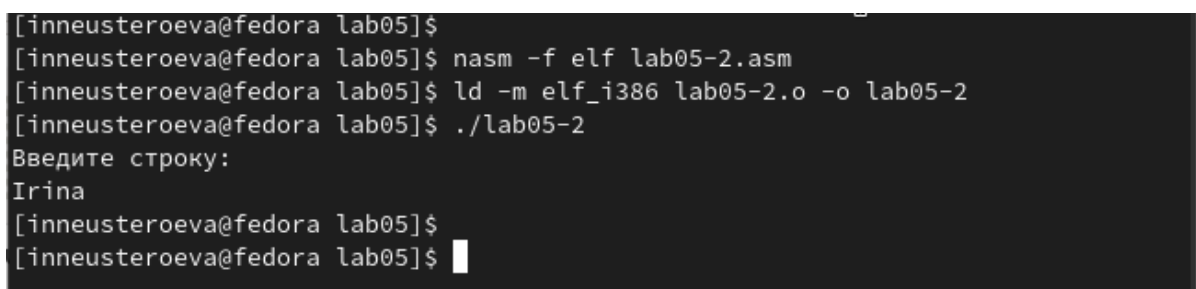
3 Написала код программы lab05-2.asm. (рис. [2.9])



```
lab05-2.asm [----] 11 L:[ 1+11 12/ 15] *(194 /
#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 sprintLF
mov ecx, buf1
mov edx, 80
call sread
call quit
```

Рис. 2.9: Программа в файле lab05-2.asm

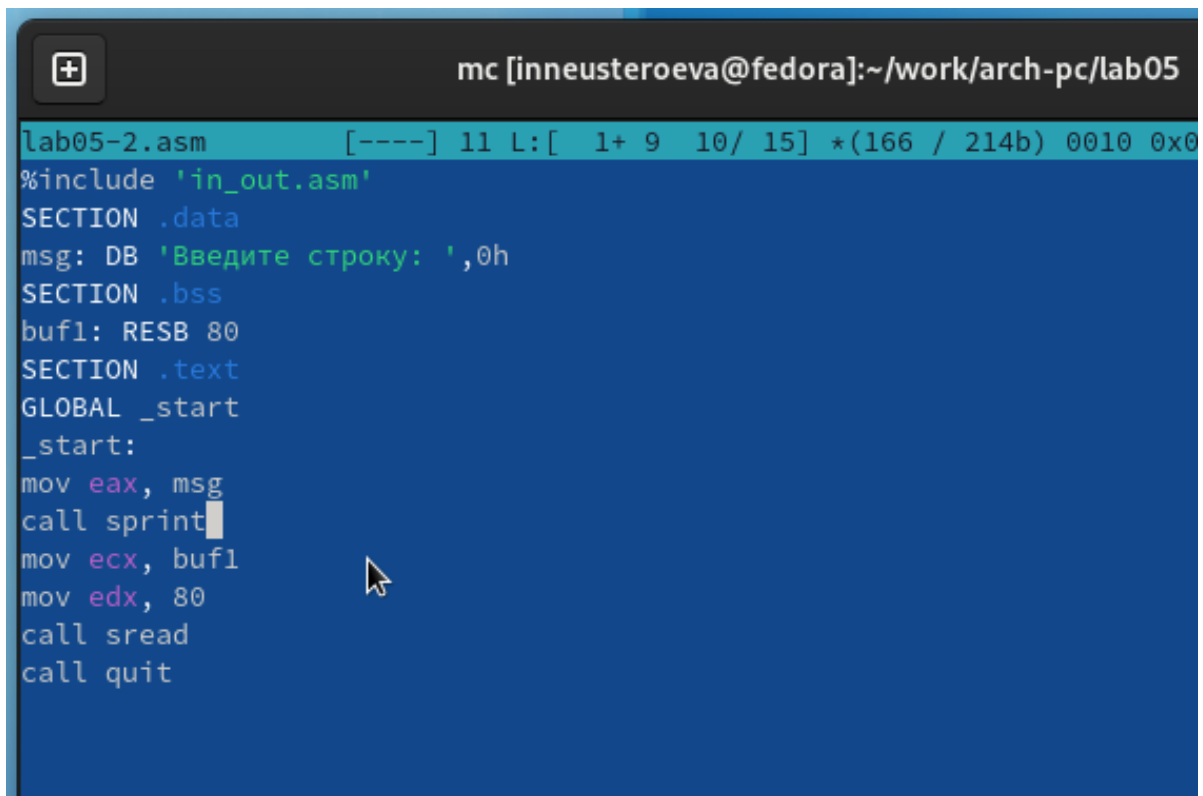
4 Скомпилировала программу и проверила запуск. (рис. [2.10])



```
[inneusteroeva@fedora lab05]$
[inneusteroeva@fedora lab05]$ nasm -f elf lab05-2.asm
[inneusteroeva@fedora lab05]$ ld -m elf_i386 lab05-2.o -o lab05-2
[inneusteroeva@fedora lab05]$ ./lab05-2
Введите строку:
Irina
[inneusteroeva@fedora lab05]$
[inneusteroeva@fedora lab05]$
```

Рис. 2.10: Запуск программы lab05-2.asm

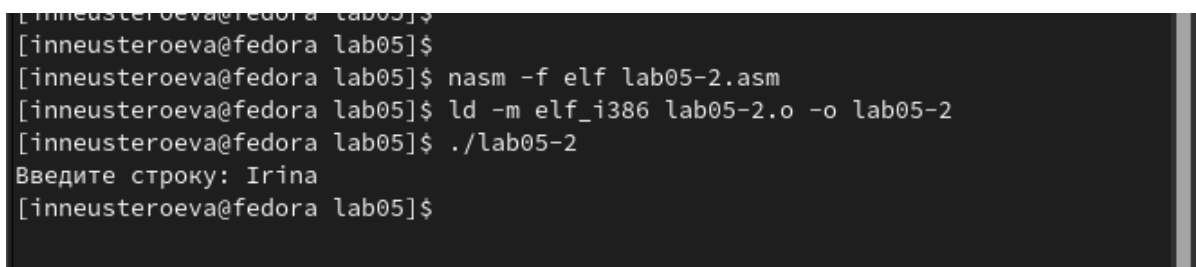
5 В файле lab5-2.asm заменила подпрограмму sprintLF на sprint (рис. [2.11]).



```
mc [inneusteroeva@fedora]:~/work/arch-pc/lab05
lab05-2.asm [-----] 11 L: [ 1+ 9 10/ 15] *(166 / 214b) 0010 0x0
#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. 2.11: Программа в файле lab05-2.asm

6 Затем я снова собрала исполняемый файл (рис. [2.12]). Теперь после вывода строки она не завершается символом перехода на новую строку.

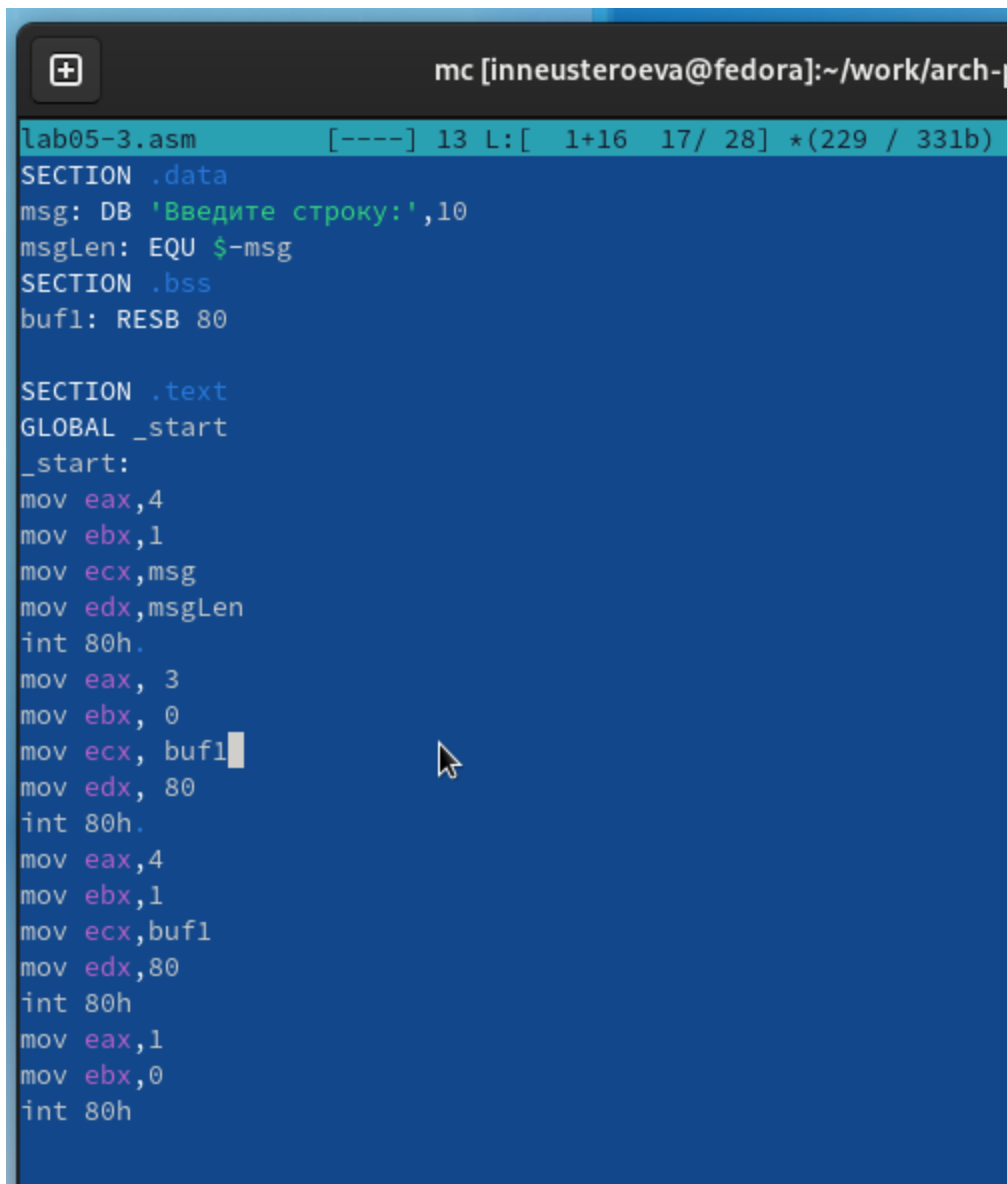


```
[inneusteroeva@fedora lab05]$
[inneusteroeva@fedora lab05]$ nasm -f elf lab05-2.asm
[inneusteroeva@fedora lab05]$ ld -m elf_i386 lab05-2.o -o lab05-2
[inneusteroeva@fedora lab05]$ ./lab05-2
Введите строку: Irina
[inneusteroeva@fedora lab05]$
```

Рис. 2.12: Запуск программы lab05-2.asm

2.3 Задания для самостоятельной работы

1 Скопировала программу lab05-1.asm и изменила код, чтобы программа выводила приглашение типа “Введите строку:”, затем считывала строку с клавиатуры и выводила введенную строку на экран. (рис. [2.13], рис. [2.14])



```
mc [inneusteroeva@fedora]:~/work/arch-p
lab05-3.asm [----] 13 L: [ 1+16 17/ 28] *(229 / 331b) (
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

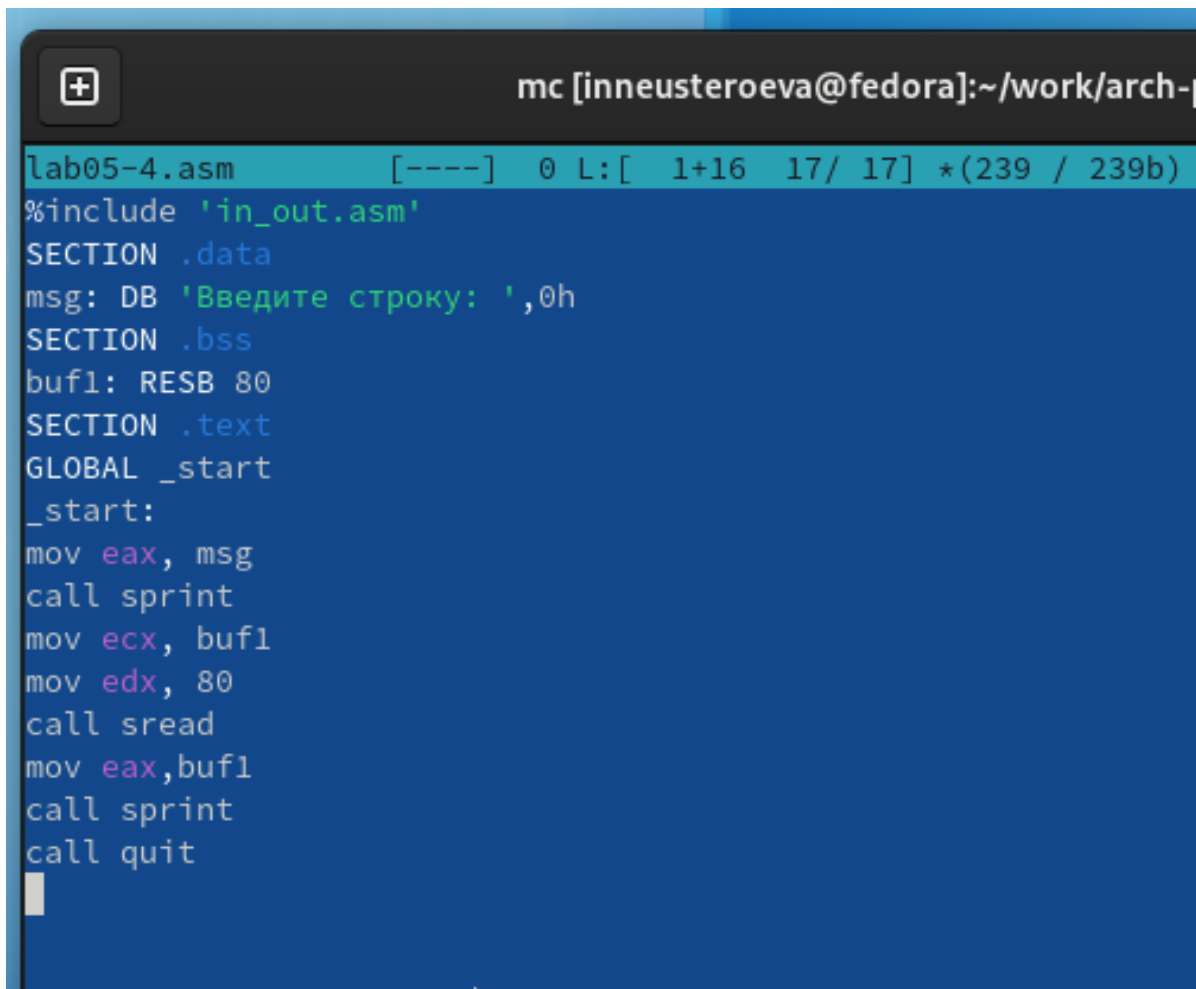
SECTION .text
GLOBAL _start
_start:
mov eax,4
mov ebx,1
mov ecx,msg
mov edx,msgLen
int 80h.
mov eax, 3
mov ebx, 0
mov ecx, buf1
mov edx, 80
int 80h.
mov eax,4
mov ebx,1
mov ecx,buf1
mov edx,80
int 80h
mov eax,1
mov ebx,0
int 80h
```

Рис. 2.13: Программа в файле lab05-3.asm

```
[inneusteroeva@fedora lab05]$  
[inneusteroeva@fedora lab05]$ nasm -f elf lab05-3.asm  
[inneusteroeva@fedora lab05]$ ld -m elf_i386 lab05-3.o -o lab05-3  
[inneusteroeva@fedora lab05]$ ./lab05-3  
Введите строку:  
Irina  
Irina  
[inneusteroeva@fedora lab05]$  
[inneusteroeva@fedora lab05]$  
[inneusteroeva@fedora lab05]$
```

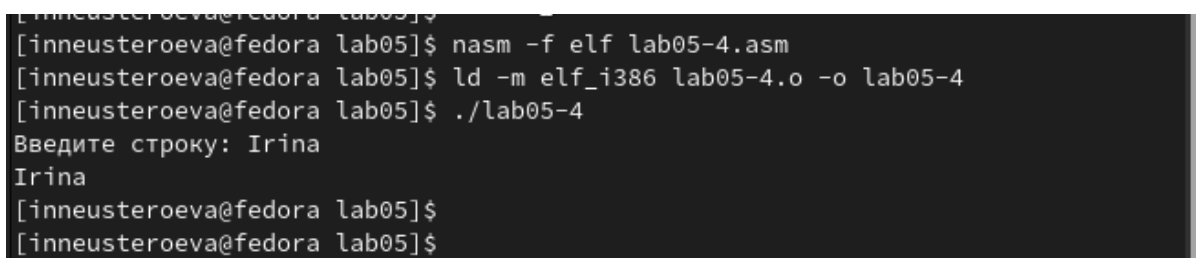
Рис. 2.14: Запуск программы lab05-3.asm

2 Также я скопировал программу lab05-2.asm и внес соответствующие изменения в код (рис. [2.15], рис. [2.16])



```
mc [inneusteroeva@fedora]:~/work/arch-  
lab05-4.asm [----] 0 L:[ 1+16 17/ 17] *(239 / 239b)  
%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  
mov eax, buf1  
call sprint  
call quit
```

Рис. 2.15: Программа в файле lab05-4.asm



```
[inneusteroeva@fedora lab05]$  
[inneusteroeva@fedora lab05]$ nasm -f elf lab05-4.asm  
[inneusteroeva@fedora lab05]$ ld -m elf_i386 lab05-4.o -o lab05-4  
[inneusteroeva@fedora lab05]$ ./lab05-4  
Введите строку: Irina  
Irina  
[inneusteroeva@fedora lab05]$  
[inneusteroeva@fedora lab05]$
```

Рис. 2.16: Запуск программы lab05-4.asm

Отличие этих двух реализаций заключается в том, что файл `in_out.asm` содержит уже готовые подпрограммы для обеспечения ввода/вывода. Таким образом,

нам остается только разместить данные в нужных регистрах и вызвать желаемую подпрограмму с помощью инструкции `call`.

3 Выводы

Научилась писать базовые ассемблерные программы. Освоила ассемблерные инструкции `mov` и `int`.