

Отчёт по лабораторной работе 5

дисциплина: Архитектура компьютера

Философова Милана

Содержание

1 Цель работы	5
2 Выводы	16

Список иллюстраций

1.1 Создание каталога lab05	5
1.2 Создание файла lab05-1.asm	6
1.3 Программа в файле lab05-1.asm	7
1.4 Просмотр файла lab05-1.asm	8
1.5 Запуск программы lab05-1.asm	9
1.6 Копирование файла lab05-1.asm в lab05-2.asm	9
1.7 Программа в файле lab05-2.asm	10
1.8 Запуск программы lab05-2.asm	10
1.9 Программа с подпрограммой sprint в файле lab05-2.asm	11
1.10 Запуск программы lab05-2.asm с изменённой подпрограммой	11
1.11 Программа в файле lab05-3.asm	12
1.12 Запуск программы lab05-3.asm	13
1.13 Программа в файле lab05-4.asm	14
1.14 Запуск программы lab05-4.asm	14

Список таблиц

1 Цель работы

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

1. Открыла **Midnight Commander**.
2. Перешла в каталог ~/work/arch-pc.
3. Создала каталог lab05.

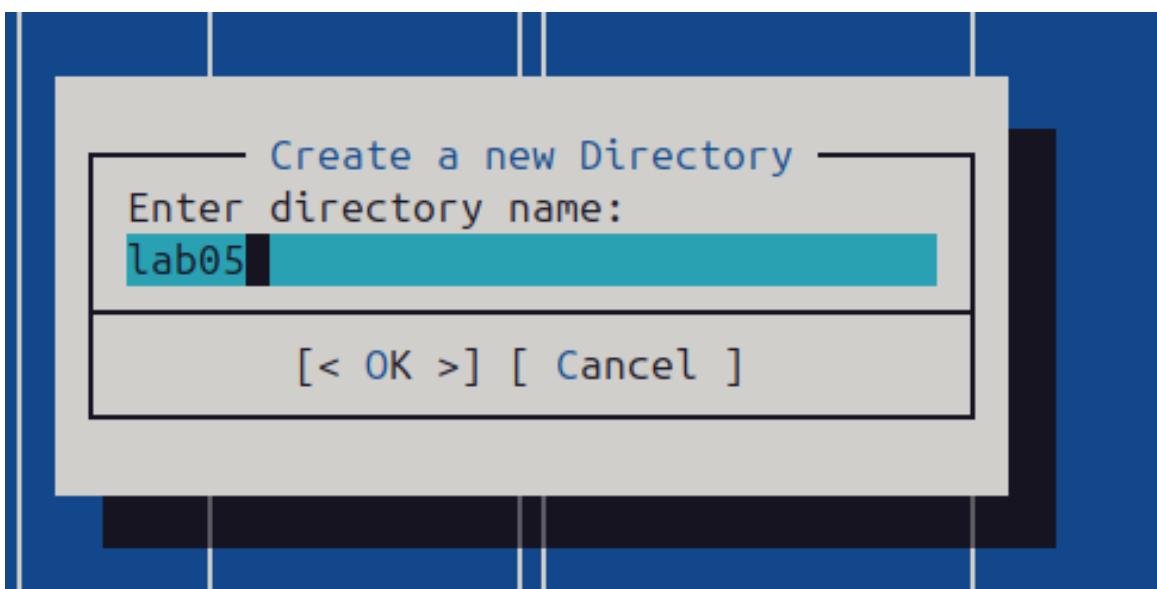


Рисунок 1.1: Создание каталога lab05

4. Создала файл lab05-1.asm.

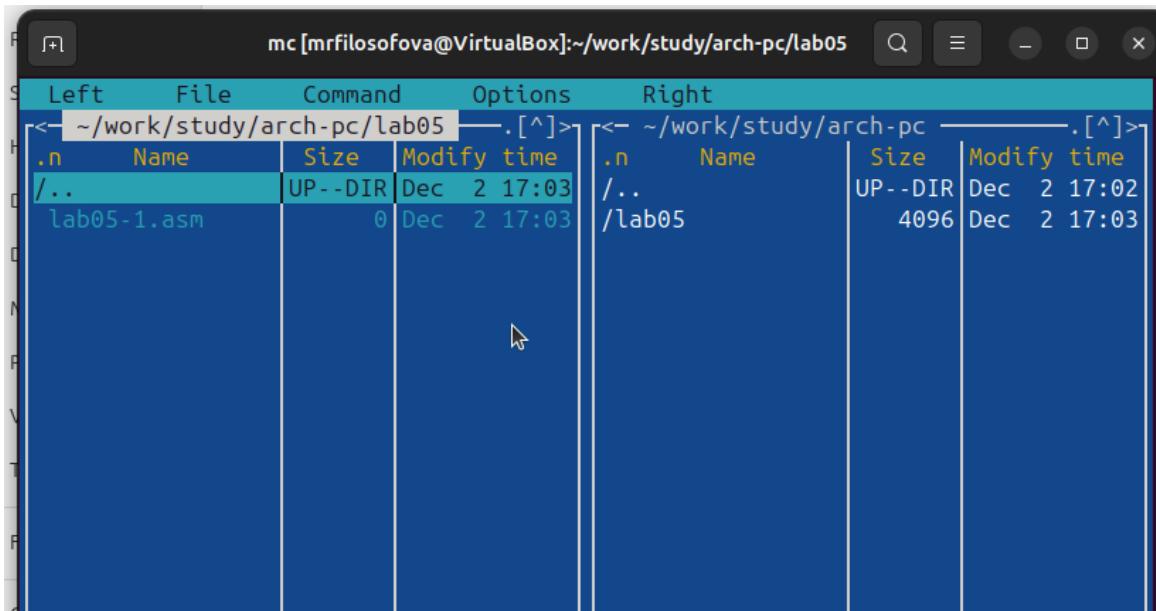
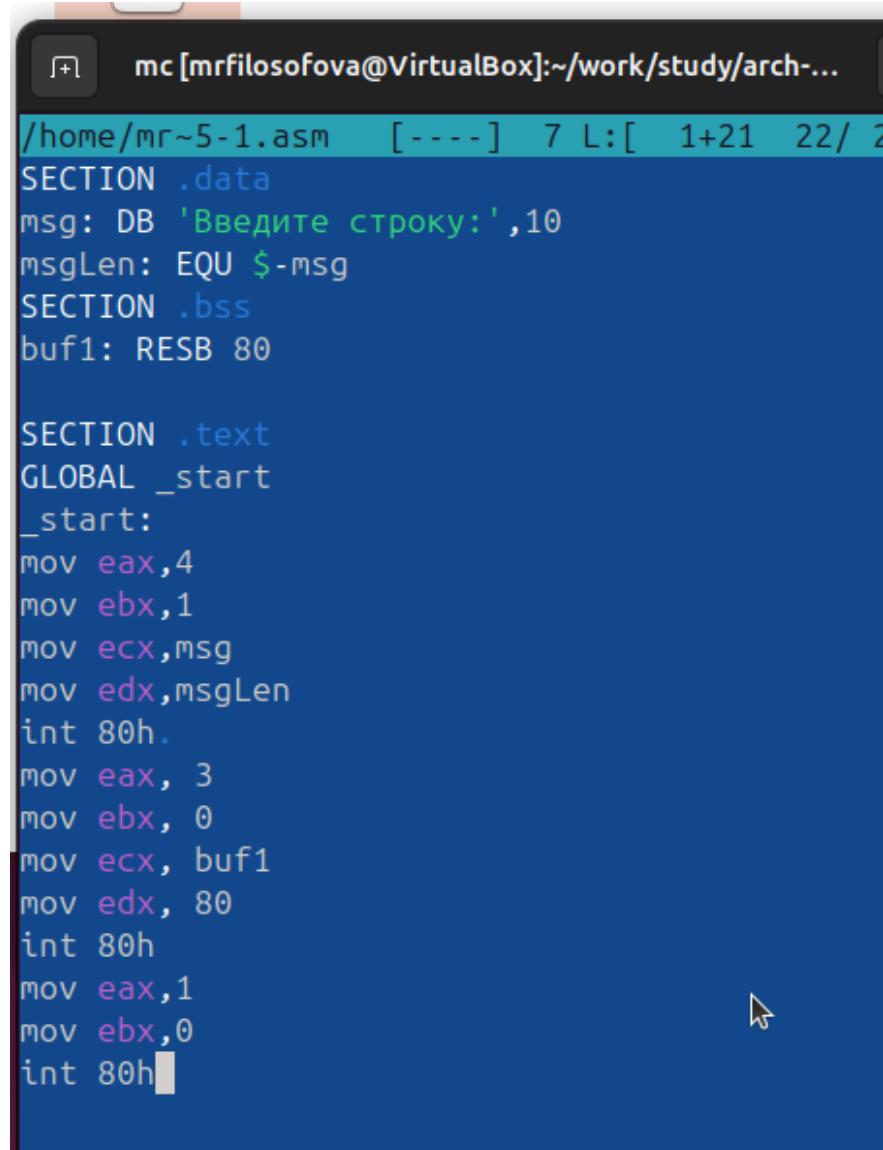


Рисунок 1.2: Создание файла lab05-1.asm

5. Открыла файл на редактирование.
 6. Написала код программы.

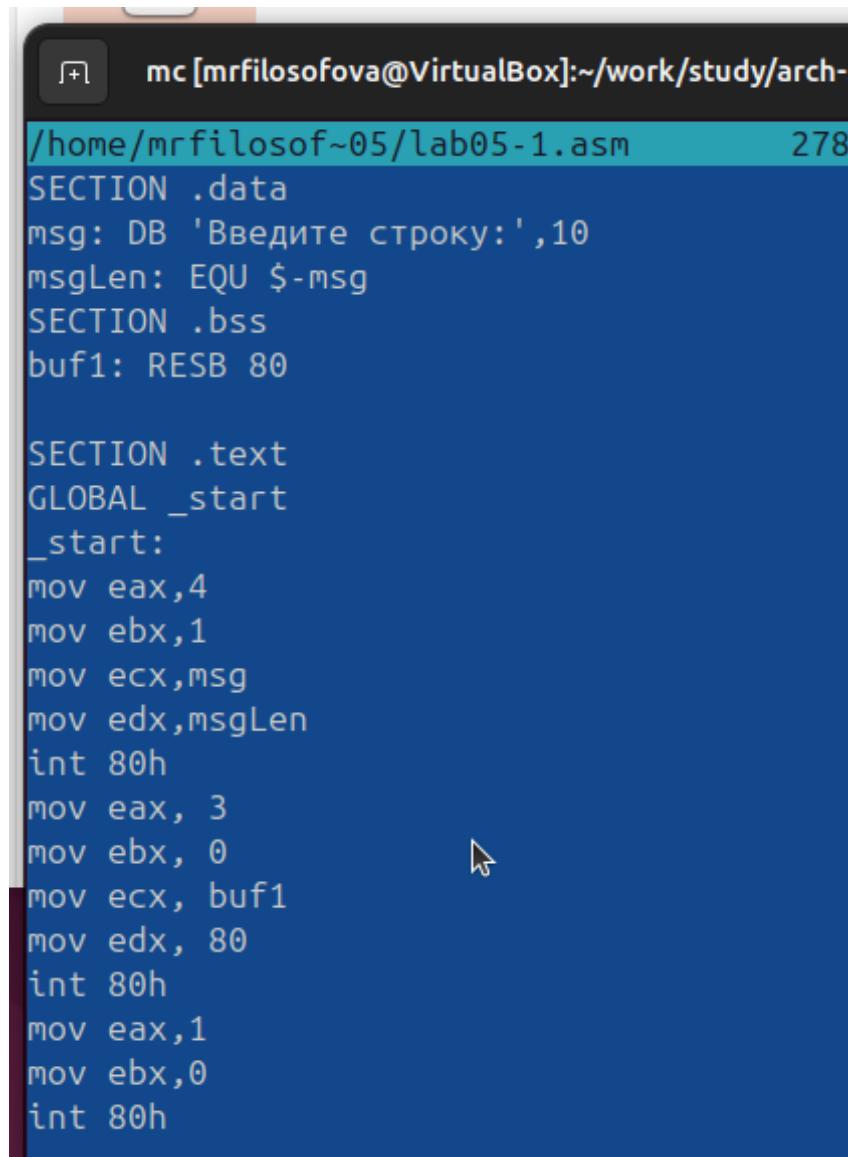


```
mc [mrilosofova@VirtualBox:~/work/study/arch-...]
/home/mr~5-1.asm  [ ---- ]  7 L:[  1+21  22/ 2
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

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
```

Рисунок 1.3: Программа в файле lab05-1.asm

7. Просмотрела содержимое файла и убедилась, что код записан корректно.



The screenshot shows a terminal window with the command 'mc' running on a VirtualBox machine. The current directory is '/home/mrfilosof~05'. The file being viewed is 'lab05-1.asm'. The assembly code is as follows:

```
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
```

Рисунок 1.4: Просмотр файла lab05-1.asm

8. Скомпилировала программу, получила исполняемый файл и проверила его работу.

```
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$ nasm -f elf lab05-1.asm
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$ ld -m elf_i386 lab05-1.o -o
lab05-1
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$ ./lab05-1
Введите строку:
123
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$
```

Рисунок 1.5: Запуск программы lab05-1.asm

9. Скачала файл `in_out.asm`.
10. Добавила файл `in_out.asm` в рабочий каталог.
11. Скопировала файл `lab05-1.asm` в `lab05-2.asm`.

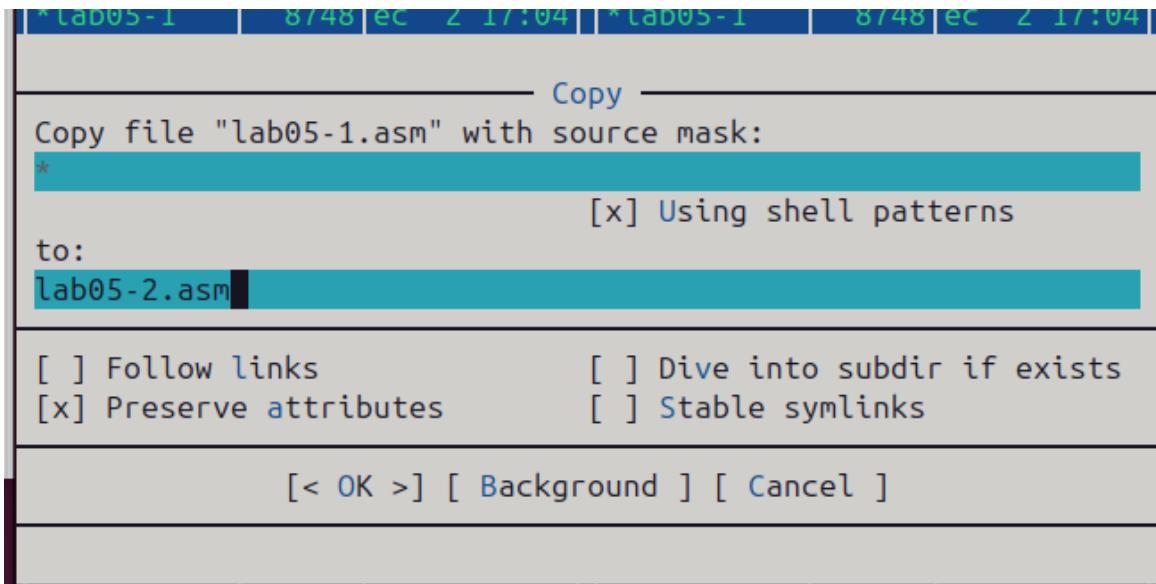
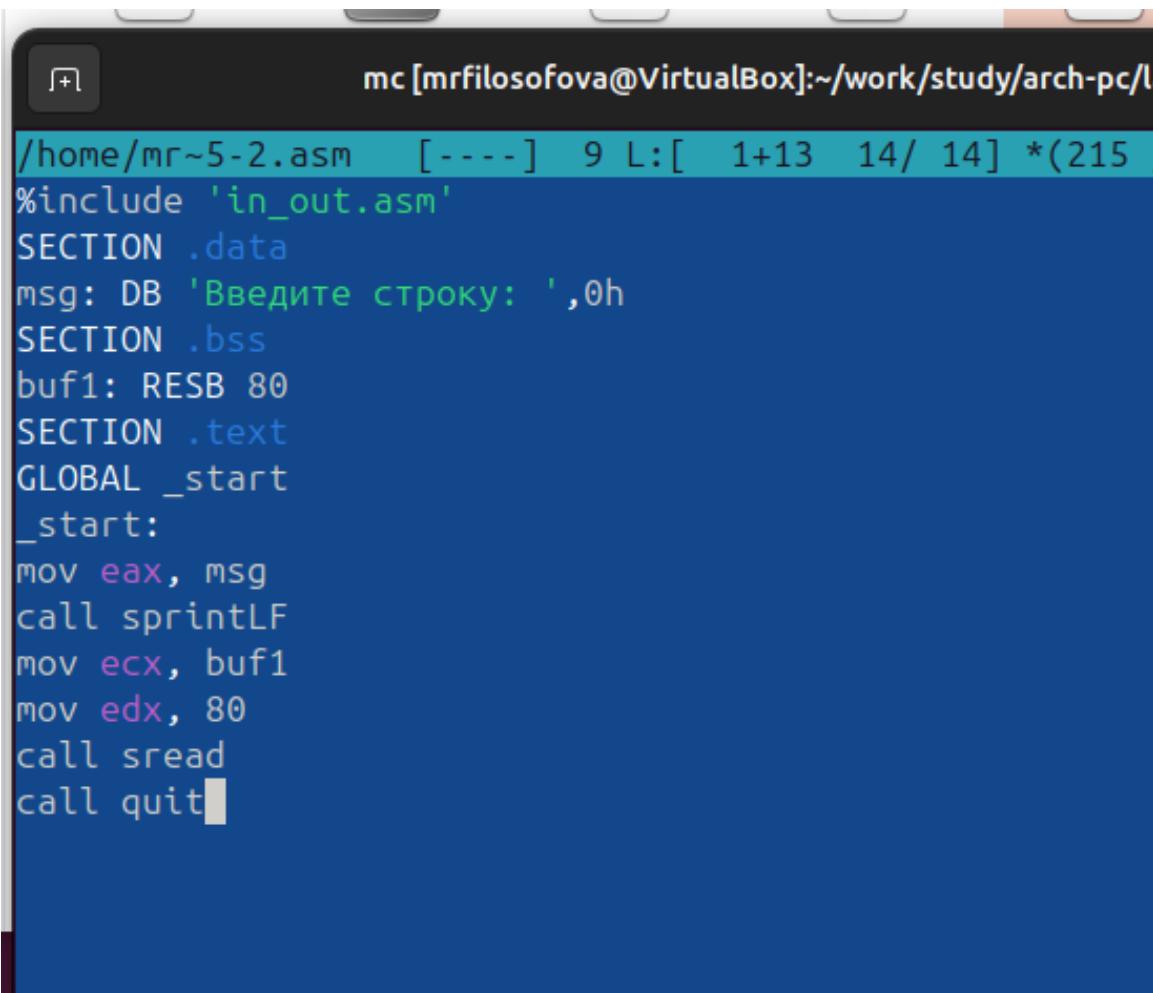


Рисунок 1.6: Копирование файла lab05-1.asm в lab05-2.asm

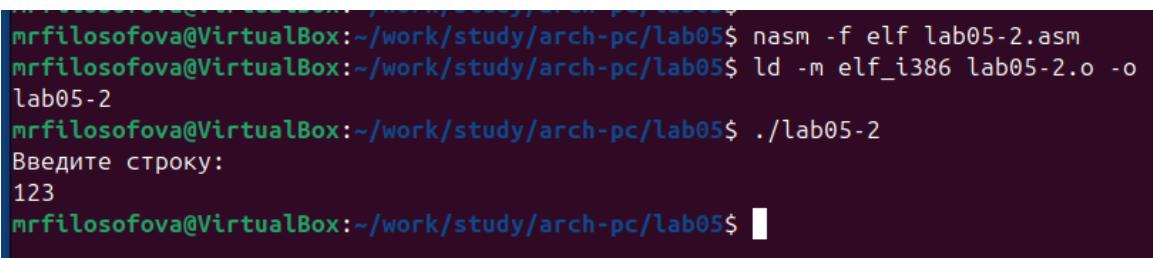
12. Написала код для программы `lab05-2.asm`, скомпилировала её и прове-рила запуск.



The screenshot shows a terminal window titled 'mc [mrilosofova@VirtualBox]:' with the command '/home/mr~5-2.asm'. The code in the editor is:

```
%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
```

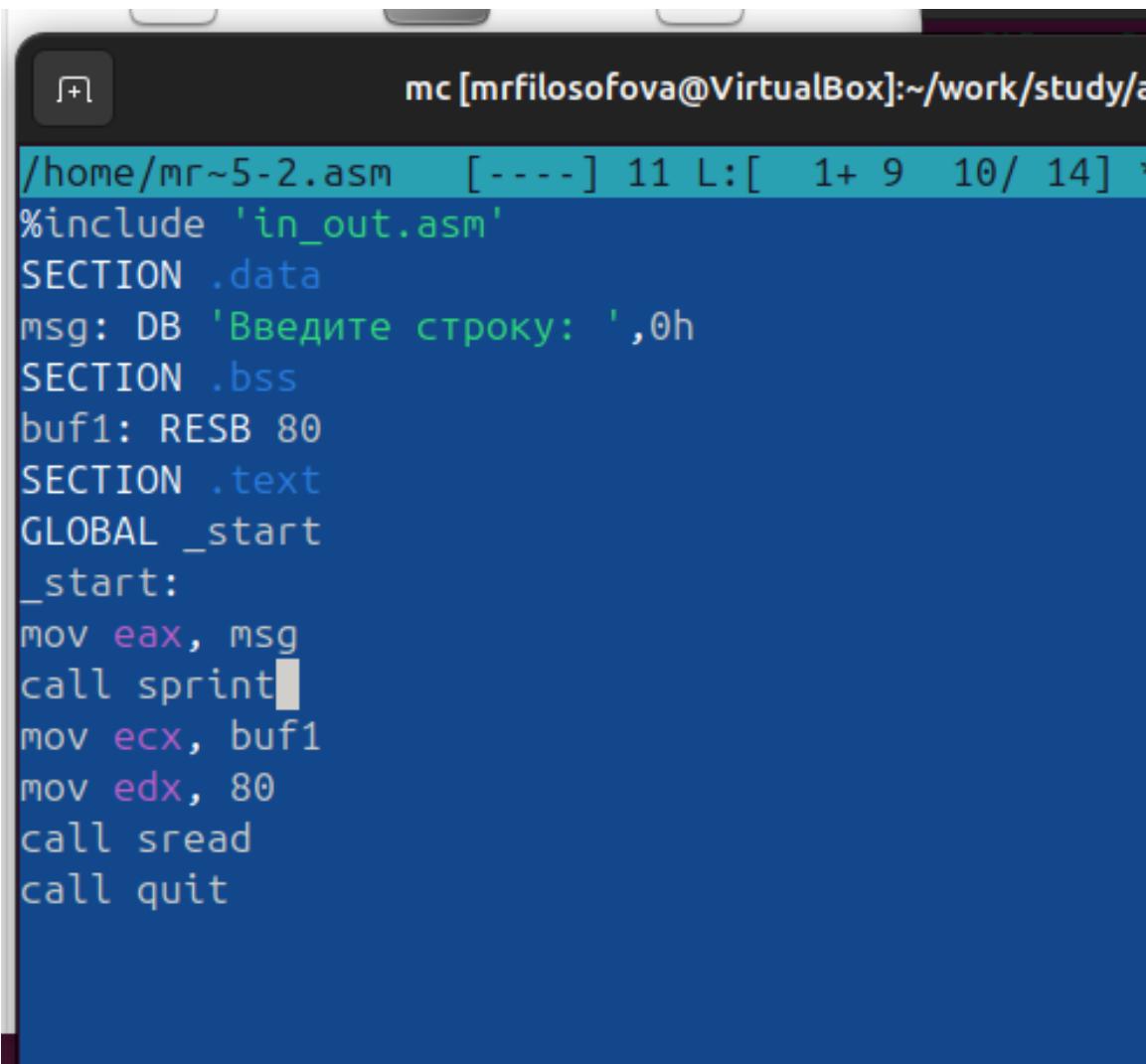
Рисунок 1.7: Программа в файле lab05-2.asm



```
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$ nasm -f elf lab05-2.asm
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$ ld -m elf_i386 lab05-2.o -o lab05-2
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$ ./lab05-2
Введите строку:
123
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$
```

Рисунок 1.8: Запуск программы lab05-2.asm

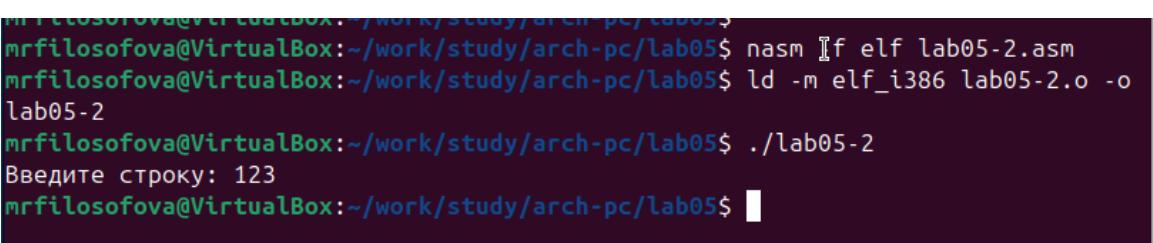
13. В программе lab05-2.asm заменила подпрограмму `sprintLF` на `sprint`.
Пересобрала исполняемый файл. Теперь вывод строки не завершается переходом на новую строку.



The screenshot shows a terminal window titled 'mc [mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05]\$'. The code displayed is:

```
/home/mr~5-2.asm [----] 11 L:[ 1+ 9 10/ 14] *
%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
```

Рисунок 1.9: Программа с подпрограммой sprint в файле lab05-2.asm

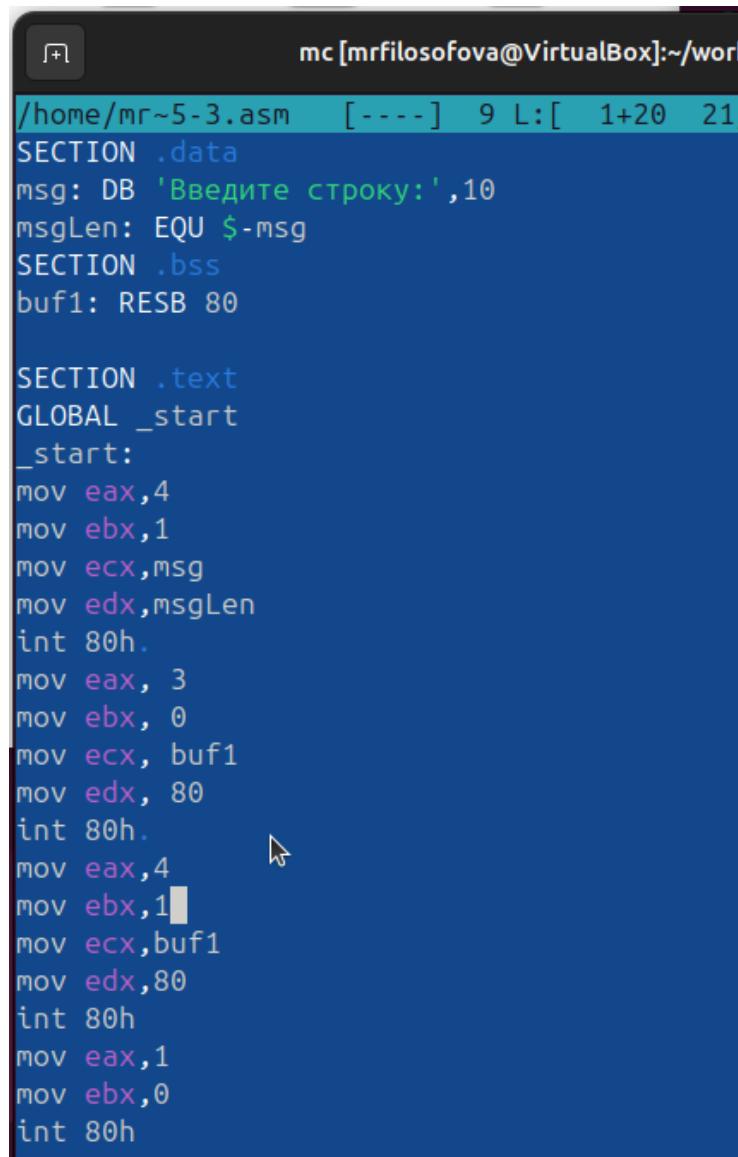


```
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$ nasm lf elf lab05-2.asm
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$ ld -m elf_i386 lab05-2.o -o lab05-2
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$ ./lab05-2
Введите строку: 123
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$
```

Рисунок 1.10: Запуск программы lab05-2.asm с изменённой подпрограммой

14. Скопировала программу lab05-1.asm и изменила код для выполнения следующих действий:

- вывод приглашения вида «Введите строку:»
- ввод строки с клавиатуры
- вывод введённой строки на экран



```

mc [mrfilosofova@VirtualBox]:~/work
/home/mr~5-3.asm [---] 9 L:[ 1+20 21
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

```

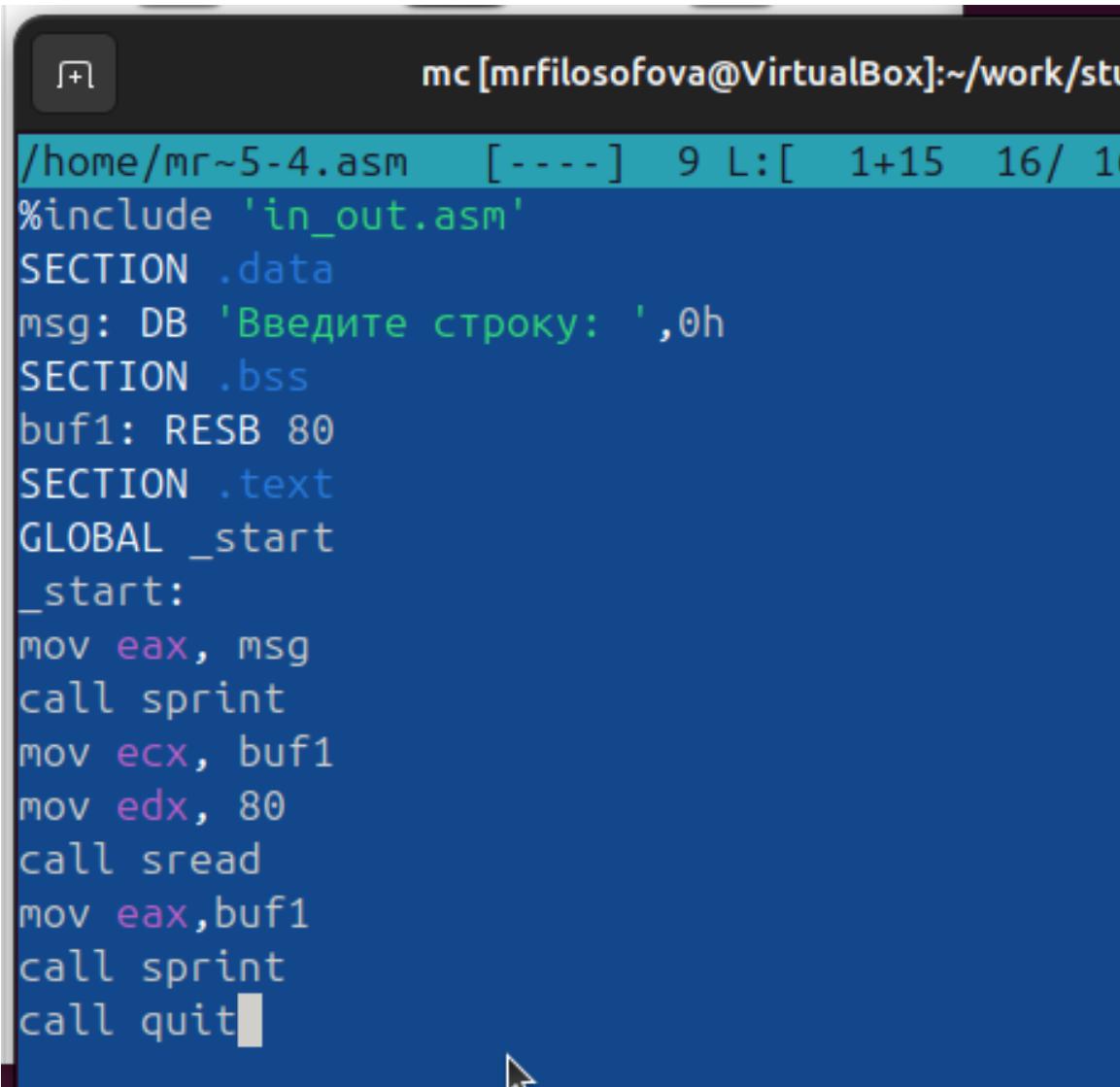
Рисунок 1.11: Программа в файле lab05-3.asm

```
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$ nasm -f elf lab05-3.asm
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$ ld -m elf_i386 lab05-3.o -o
lab05-3
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$ ./lab05-3
Введите строку:
123
123
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$
```

Рисунок 1.12: Запуск программы lab05-3.asm

15. Скопировала программу lab05-2.asm и внесла изменения для выполнения аналогичных действий:

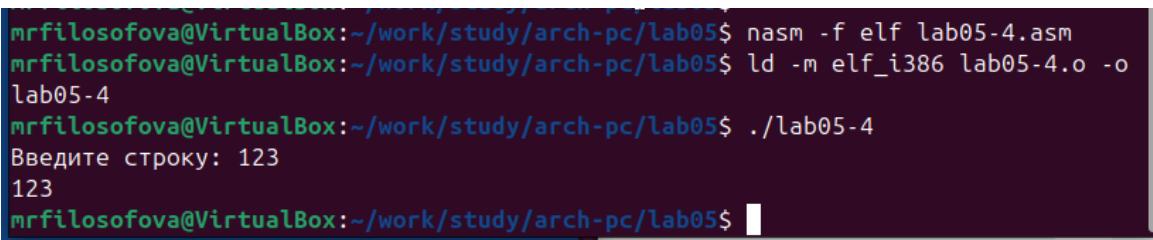
- вывод приглашения вида «Введите строку:»
- ввод строки с клавиатуры
- вывод введённой строки на экран



The screenshot shows a terminal window titled 'mc [mrfilosofova@VirtualBox]:~/work/stu'. The code displayed is:

```
/home/mr~5-4.asm [----] 9 L:[ 1+15 16/ 16
%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
```

Рисунок 1.13: Программа в файле lab05-4.asm



```
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$ nasm -f elf lab05-4.asm
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$ ld -m elf_i386 lab05-4.o -o lab05-4
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$ ./lab05-4
Введите строку: 123
123
mrfilosofova@VirtualBox:~/work/study/arch-pc/lab05$
```

Рисунок 1.14: Запуск программы lab05-4.asm

Отличие двух реализаций: В реализации на основе файла `in_out.asm` ис-

пользуются готовые подпрограммы для ввода и вывода. Это позволяет сосредоточиться только на размещении данных в нужных регистрах и вызове подпрограмм с помощью инструкции call.

2 Выводы

Научились писать базовые ассемблерные программы. Освоили ассемблерные инструкции mov и int.