

# **Отчёт по лабораторной работе 5**

**Архитектура компьютера**

Артемов Данил, НБИбд-01-24, 1132249998

# Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	15

## Список иллюстраций

2.1	Создание каталога . . . . .	6
2.2	Код lab05-1.asm . . . . .	7
2.3	Просмотр файла lab05-1.asm . . . . .	8
2.4	Запуск программы lab05-1.asm . . . . .	9
2.5	Копирование файла . . . . .	9
2.6	Код lab05-2.asm . . . . .	10
2.7	Запуск программы lab05-2.asm . . . . .	10
2.8	Код lab05-2.asm . . . . .	11
2.9	Запуск программы lab05-2.asm . . . . .	11
2.10	Код lab05-3.asm . . . . .	12
2.11	Запуск программы lab05-3.asm . . . . .	13
2.12	Код lab05-4.asm . . . . .	14
2.13	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`.

## 2 Выполнение лабораторной работы

1. Я открыл Midnight Commander и перешел в каталог ~/work/arch-pc. Создал новый каталог под названием lab05.

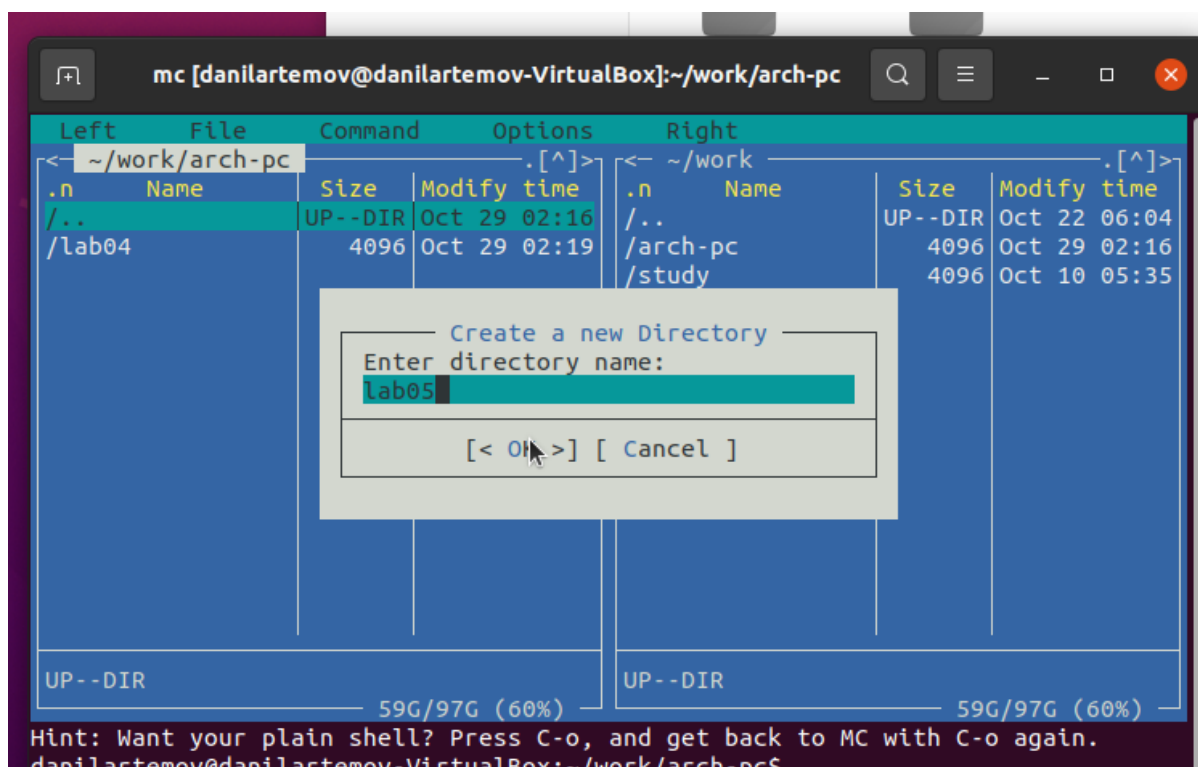


Рис. 2.1: Создание каталога

2. Внутри каталога lab05 я создал файл lab05-1.asm. Затем я открыл этот файл для редактирования и начал писать код.



```
danilartemov@danilartemov-  
/home/dan~b05-1.asm [ - - - - ] 7 L:[ 1+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,1  
mov ebx,0  
int 80h
```

Рис. 2.2: Код lab05-1.asm

3. Я также открыл файл lab05-1.asm для просмотра и проверил, что код был написан правильно.



```
mc [danilartemov@danilartemov-VirtualB
/home/danilartemov/work/~ch-pc/lab05/lab
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. 2.3: Просмотр файла lab05-1.asm

4. После того, как я получил исполняемый файл из кода, я проверил его работу, чтобы убедиться, что все функционирует должным образом.



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

Рис. 2.4: Запуск программы lab05-1.asm

5. Я скачал файл in\_out.asm и добавил его в рабочий каталог. Затем я скопировал содержимое файла lab05-1.asm в новый файл под названием lab05-2.asm.

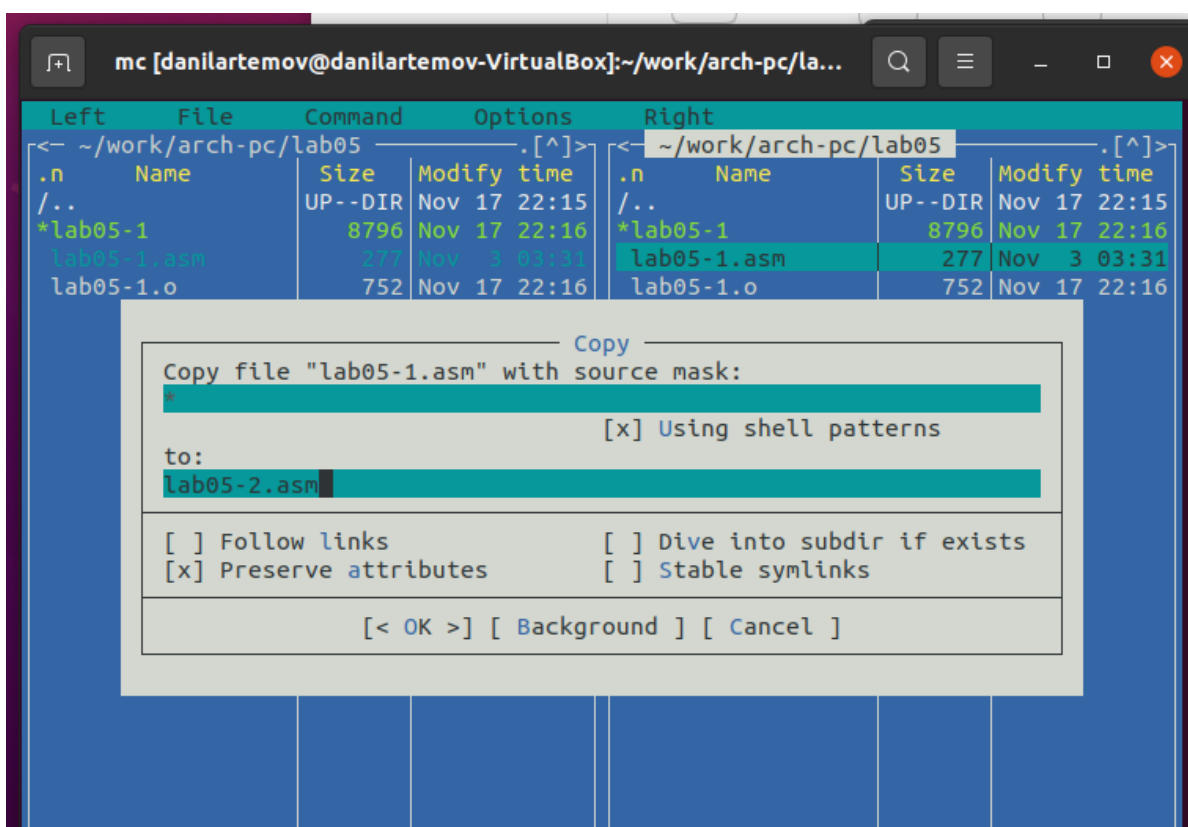
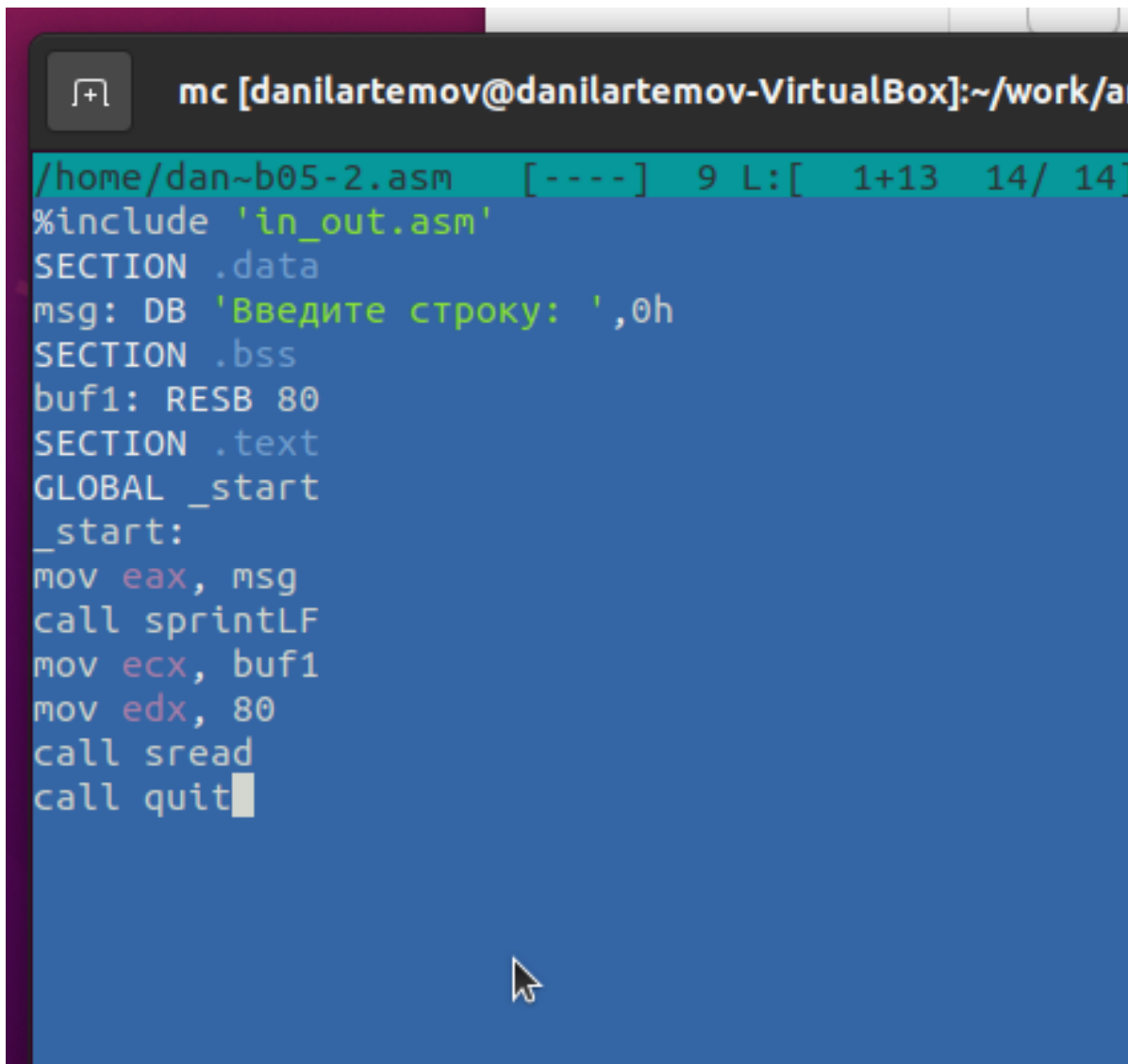


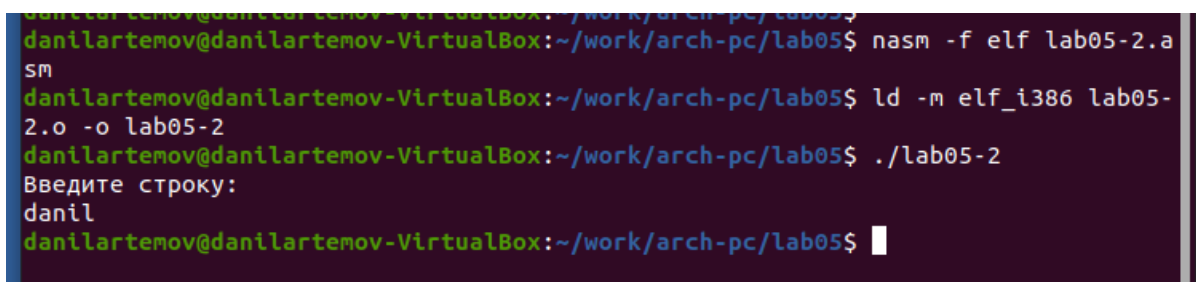
Рис. 2.5: Копирование файла

6. В файле lab05-2.asm я написал код программы. После этого я скомпилировал программу и проверил ее запуск.



```
mc [danilartemov@danilartemov-VirtualBox]:~/work/a
/home/dan~b05-2.asm  [ - - - - ]  9 L:[  1+13  14/ 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 sprintLF
mov ecx, buf1
mov edx, 80
call sread
call quit
```

Рис. 2.6: Код lab05-2.asm

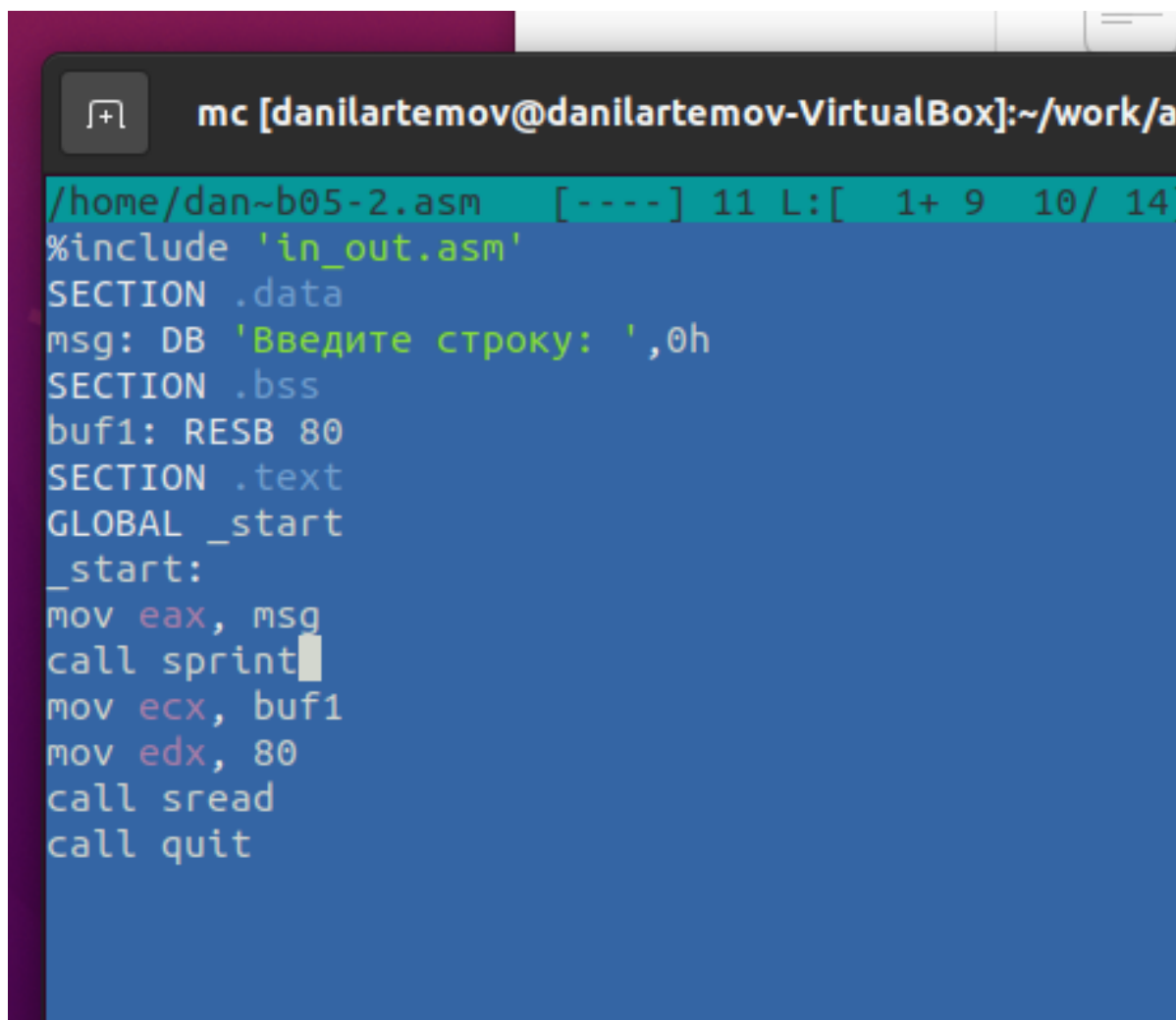


```
danilartemov@danilartemov-VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab05-2.a
sm
danilartemov@danilartemov-VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 lab05-
2.o -o lab05-2
danilartemov@danilartemov-VirtualBox:~/work/arch-pc/lab05$ ./lab05-2
Введите строку:
danil
danilartemov@danilartemov-VirtualBox:~/work/arch-pc/lab05$
```

Рис. 2.7: Запуск программы lab05-2.asm

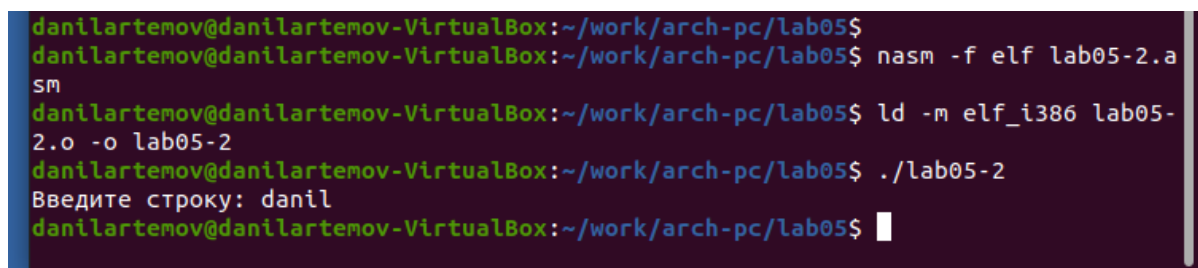
7. Внутри файла lab05-2.asm я заменил подпрограмму sprintLF на sprint. Затем

я пересобрал исполняемый файл. Теперь при выводе строки нет перехода на следующую строку.



```
mc [danilartemov@danilartemov-VirtualBox]:~/work/a
/home/dan~b05-2.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
```

Рис. 2.8: Код lab05-2.asm



```
danilartemov@danilartemov-VirtualBox:~/work/arch-pc/lab05$
danilartemov@danilartemov-VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab05-2.a
sm
danilartemov@danilartemov-VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 lab05-
2.o -o lab05-2
danilartemov@danilartemov-VirtualBox:~/work/arch-pc/lab05$ ./lab05-2
Введите строку: danil
danilartemov@danilartemov-VirtualBox:~/work/arch-pc/lab05$
```

Рис. 2.9: Запуск программы lab05-2.asm

8. Я скопировал программу lab05-1.asm и изменил код так, чтобы сначала выводилось приглашение “Введите строку:”, затем пользователь вводил строку с клавиатуры, и введенная строка выводилась на экран.



```
mc [danilartemov@danilartemov-Virtuale
/home/dan~b05-3.asm [---] 8 L:[ 1+
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

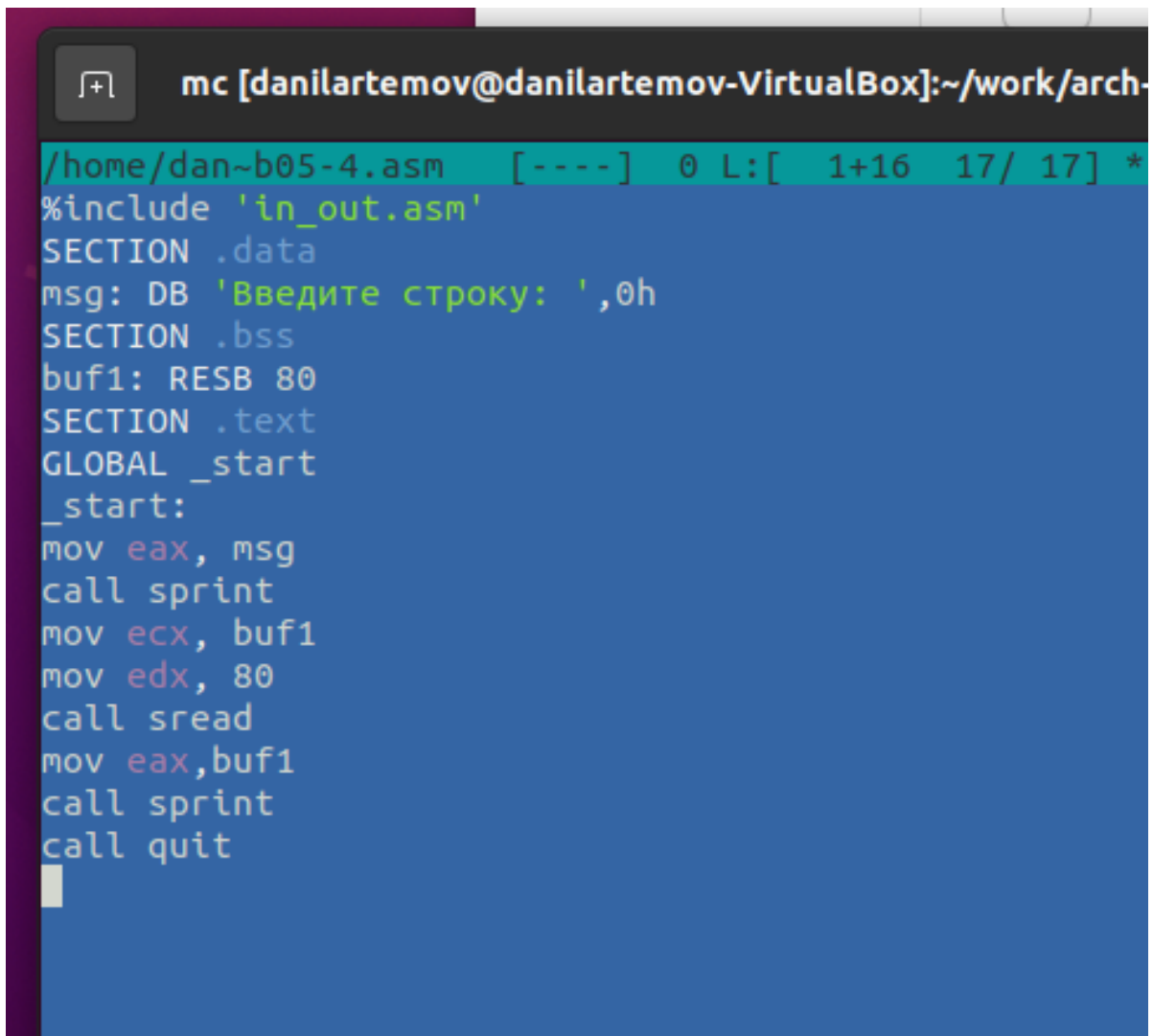
SECTION .text
GLOBAL _start
_start:
mov eax,4
mov ebx,1
mov ecx,msg
mov edx,msgLen
int 80h.
mov eax, 3
mov ebx, 0
mov ecx, buf1
mov edx, 80
int 80h.
mov eax,4
mov ebx,1
mov ecx,buf1
mov edx,80
int 80h
mov eax,1
mov ebx,0
int 80h
```

Рис. 2.10: Код lab05-3.asm

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

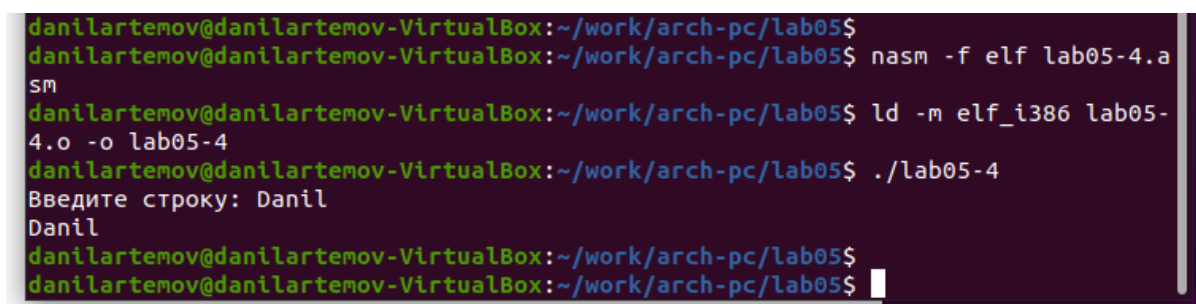
Рис. 2.11: Запуск программы lab05-3.asm

9. Затем я скопировал программу lab05-2.asm и выполнил аналогичные действия, описанные выше, но теперь использовал возможности из файла in\_out.asm.



```
mc [danilartemov@danilartemov-VirtualBox]:~/work/arch-  
/home/dan~b05-4.asm [ - - - ] 0 L:[ 1+16 17/ 17] *  
%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. 2.12: Код lab05-4.asm



```
danilartemov@danilartemov-VirtualBox:~/work/arch-pc/lab05$  
danilartemov@danilartemov-VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab05-4.a  
sm  
danilartemov@danilartemov-VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 lab05-  
4.o -o lab05-4  
danilartemov@danilartemov-VirtualBox:~/work/arch-pc/lab05$ ./lab05-4  
Введите строку: Danil  
Danil  
danilartemov@danilartemov-VirtualBox:~/work/arch-pc/lab05$  
danilartemov@danilartemov-VirtualBox:~/work/arch-pc/lab05$
```

Рис. 2.13: Запуск программы lab05-4.asm

## 3 Выводы

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