

# Отчёт по лабораторной работе №5

Архитектура компьютера

Койлюбаев Сыймык Эркинович

# Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Задание для самостоятельной работы . . . . .	13
3	Выводы	17

## Список иллюстраций

2.1	окно Midnight Commander . . . . .	6
2.2	Создание каталога . . . . .	7
2.3	touch lab05-1.asm . . . . .	7
2.4	Код программы lab05-1.asm . . . . .	8
2.5	Проверка кода lab05-1.asm . . . . .	9
2.6	Компиляция и запуск программы lab05-1.asm . . . . .	10
2.7	Копирование файла in_out.asm . . . . .	10
2.8	Копирование файла lab05-1.asm . . . . .	11
2.9	Код программы lab05-2.asm . . . . .	12
2.10	Компиляция и запуск программы lab05-2.asm . . . . .	12
2.11	Код программы lab05-2.asm . . . . .	13
2.12	Компиляция и запуск программы lab05-2.asm . . . . .	13
2.13	Код программы lab05-3.asm . . . . .	14
2.14	Компиляция и запуск программы lab05-3.asm . . . . .	15
2.15	Код программы lab05-4.asm . . . . .	16
2.16	Компиляция и 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

Я запустил файловый менеджер Midnight Commander, переместился в директорию ~/work/arch-pc с помощью стрелок и клавиши ввода, после чего нажатием F7 создал папку lab05.

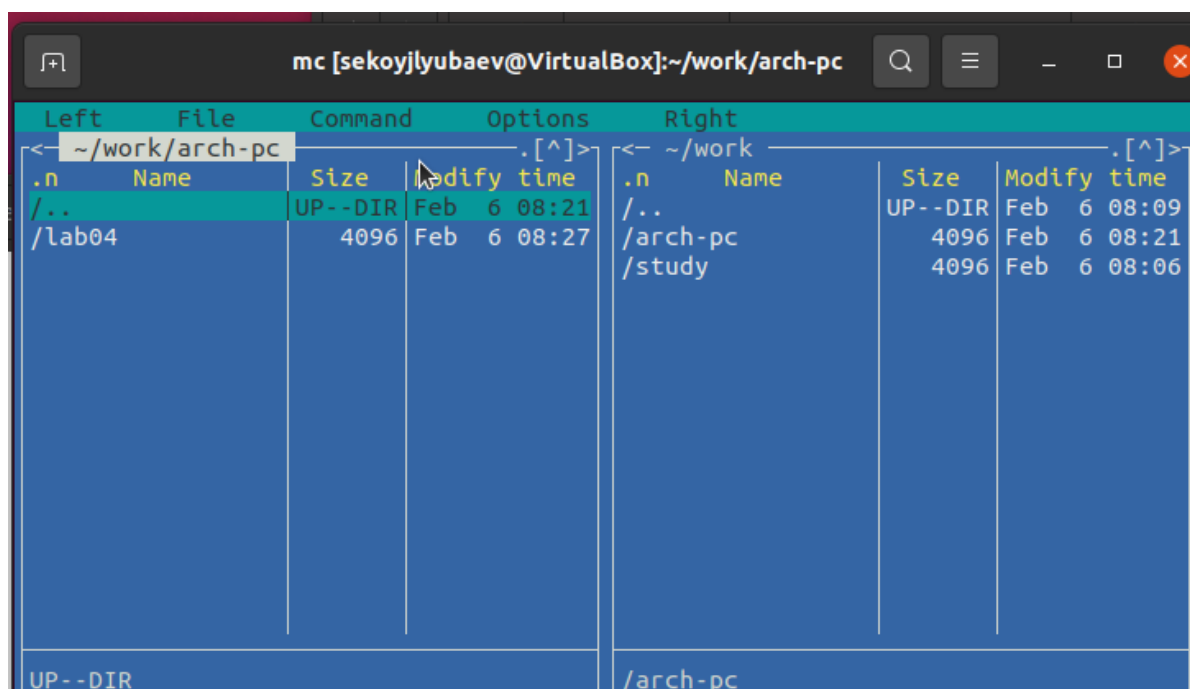


Рис. 2.1: окно Midnight Commander

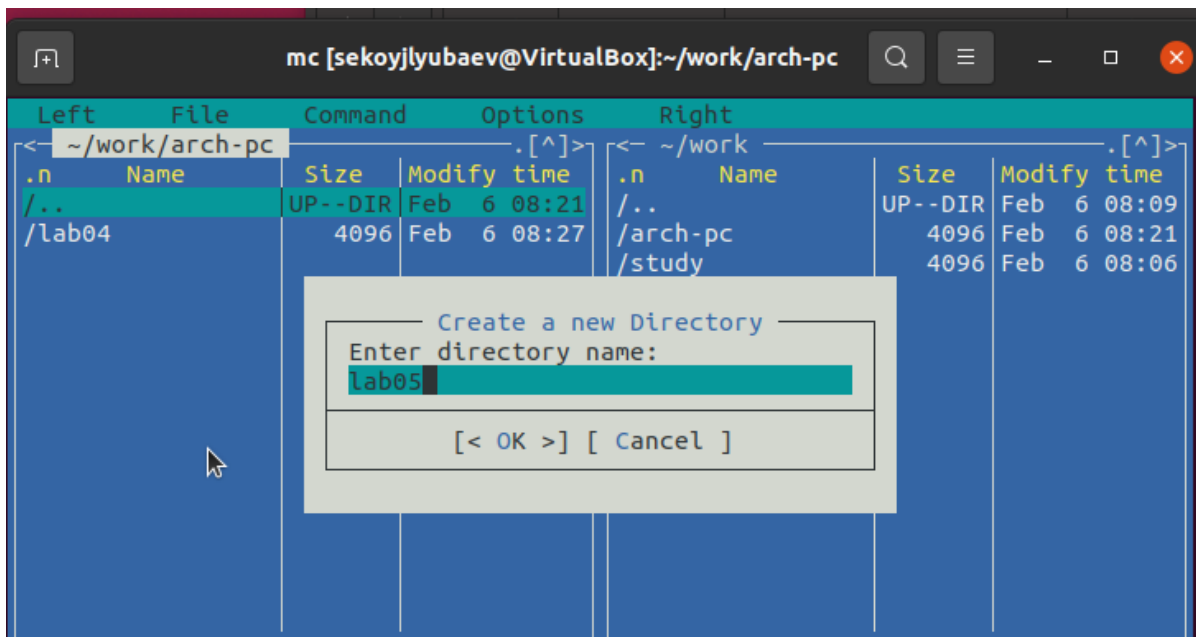


Рис. 2.2: Создание каталога

Затем, используя команду touch, я создал файл lab05-1.asm.

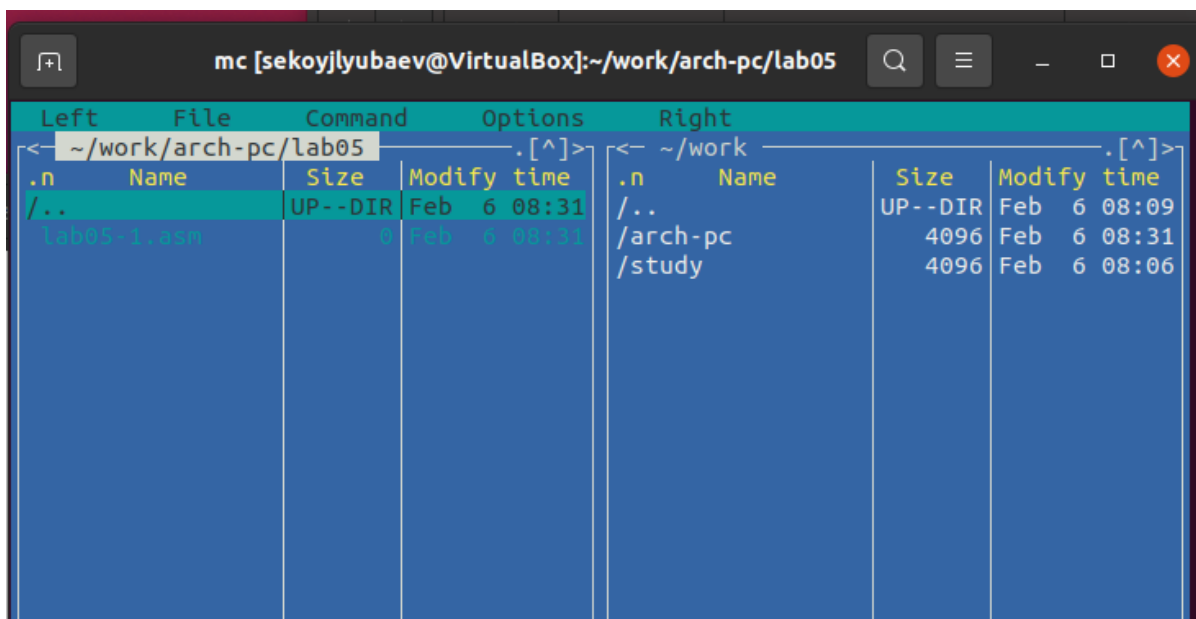
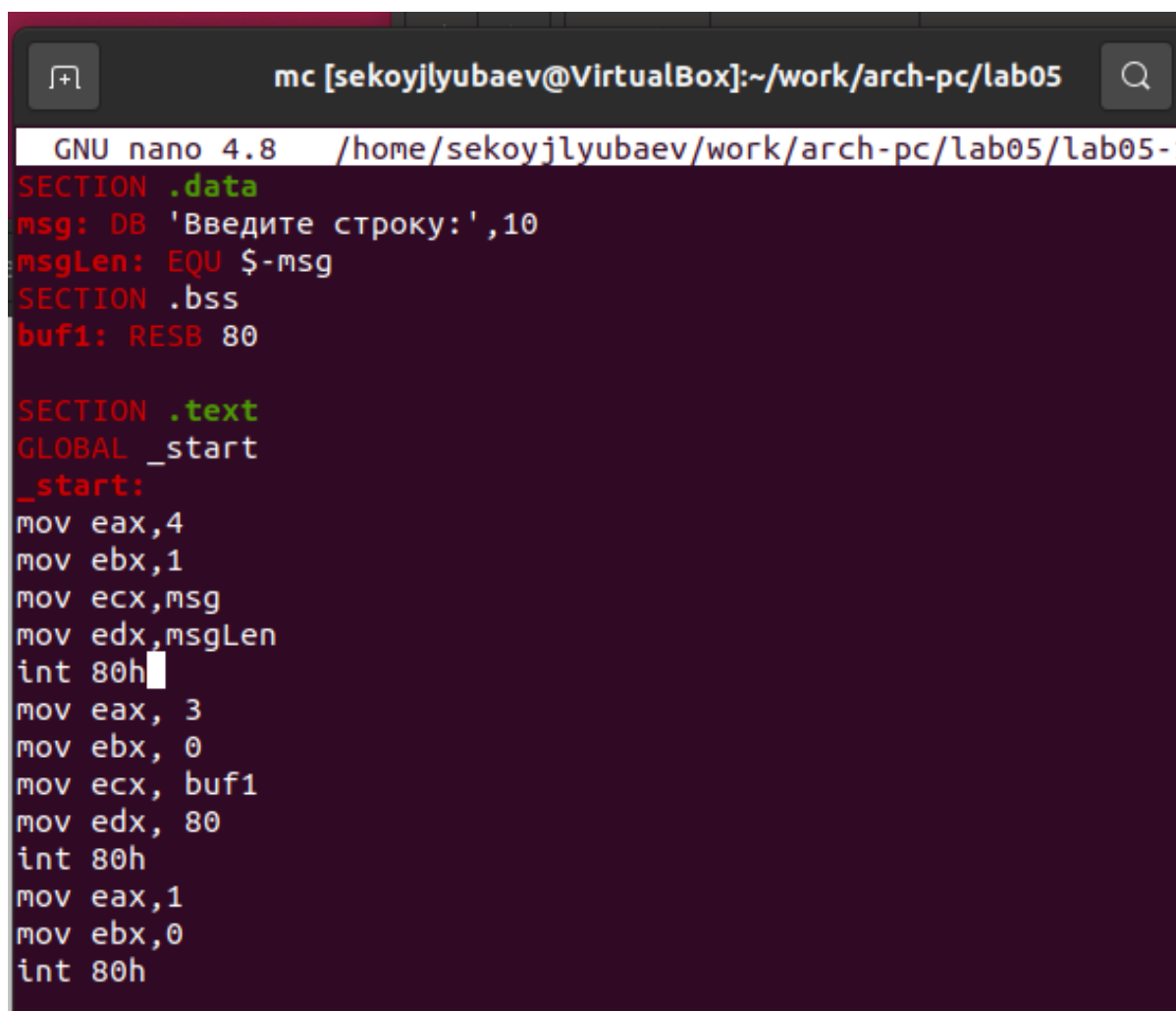


Рис. 2.3: touch lab05-1.asm

Для редактирования файла я активировал редактор mceditor, нажав F4, и при-

ступил к написанию кода, который соответствовал поставленной задаче.



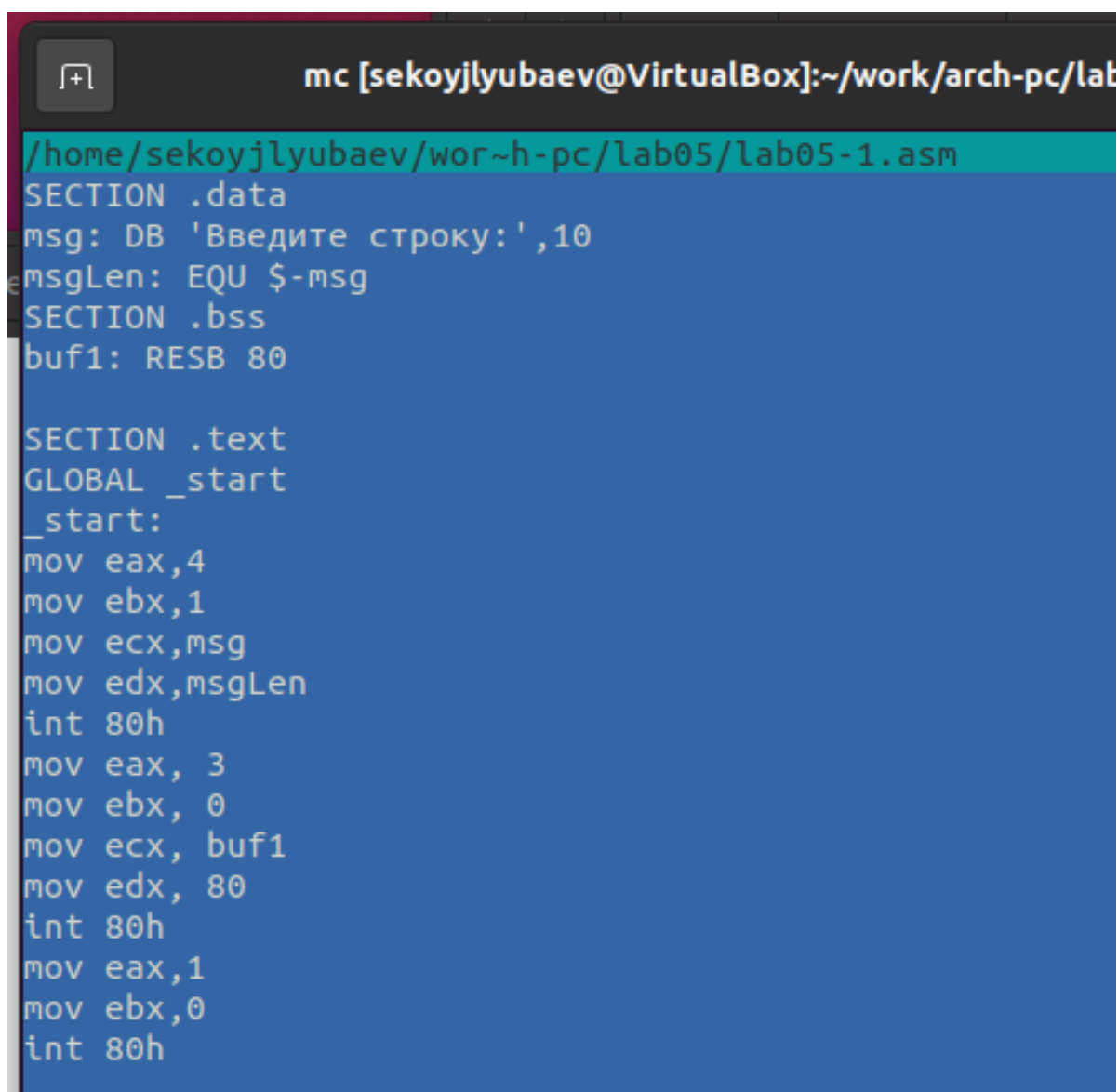
```
mc [sekoyjlyubaev@VirtualBox]:~/work/arch-pc/lab05
GNU nano 4.8 /home/sekoyjlyubaev/work/arch-pc/lab05/lab05-1.asm
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

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

Рис. 2.4: Код программы lab05-1.asm

Для проверки содержимого файла на наличие введенного кода я воспользовался просмотрщиком, вызванным клавишей F3.





```
mc [sekoyjlyubaev@VirtualBox]:~/work/arch-pc/lab
/home/sekoyjlyubaev/wor~h-pc/lab05/lab05-1.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,1
mov ebx,0
int 80h
```

Рис. 2.5: Проверка кода lab05-1.asm

Я осуществил компиляцию написанной программы, преобразовав исходный код в объектный файл и собрав исполняемый файл, чтобы проверить ее функциональность.

```

sekoyjlyubaev@VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab05-1.asm
sekoyjlyubaev@VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 lab05-1.o -o lab05-1
sekoyjlyubaev@VirtualBox:~/work/arch-pc/lab05$ ./lab05-1
Введите строку:
test
sekoyjlyubaev@VirtualBox:~/work/arch-pc/lab05$ █

```

Рис. 2.6: Компиляция и запуск программы lab05-1.asm

Загрузил файл in\_out.asm и поместил его в текущий рабочий каталог.

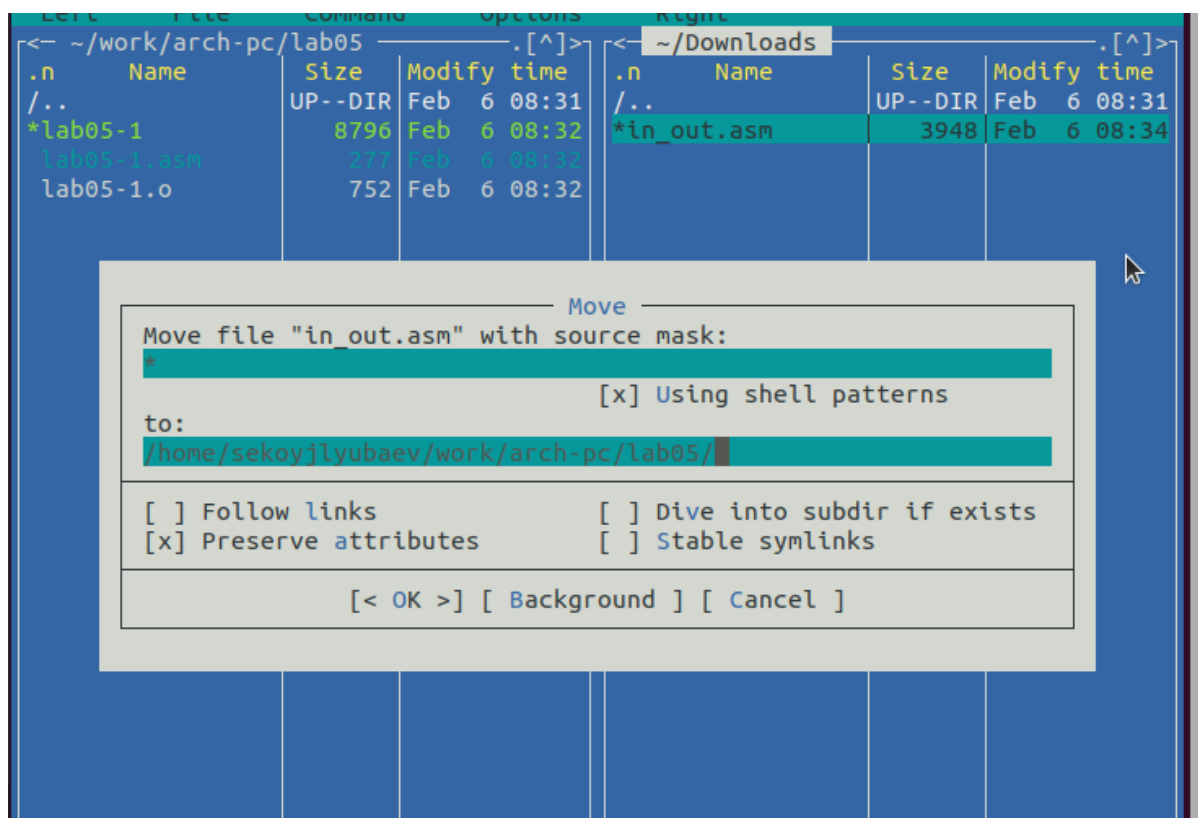


Рис. 2.7: Копирование файла in\_out.asm

Используя клавишу F5, я скопировал код из файла lab05-1.asm в новый файл lab05-2.asm.

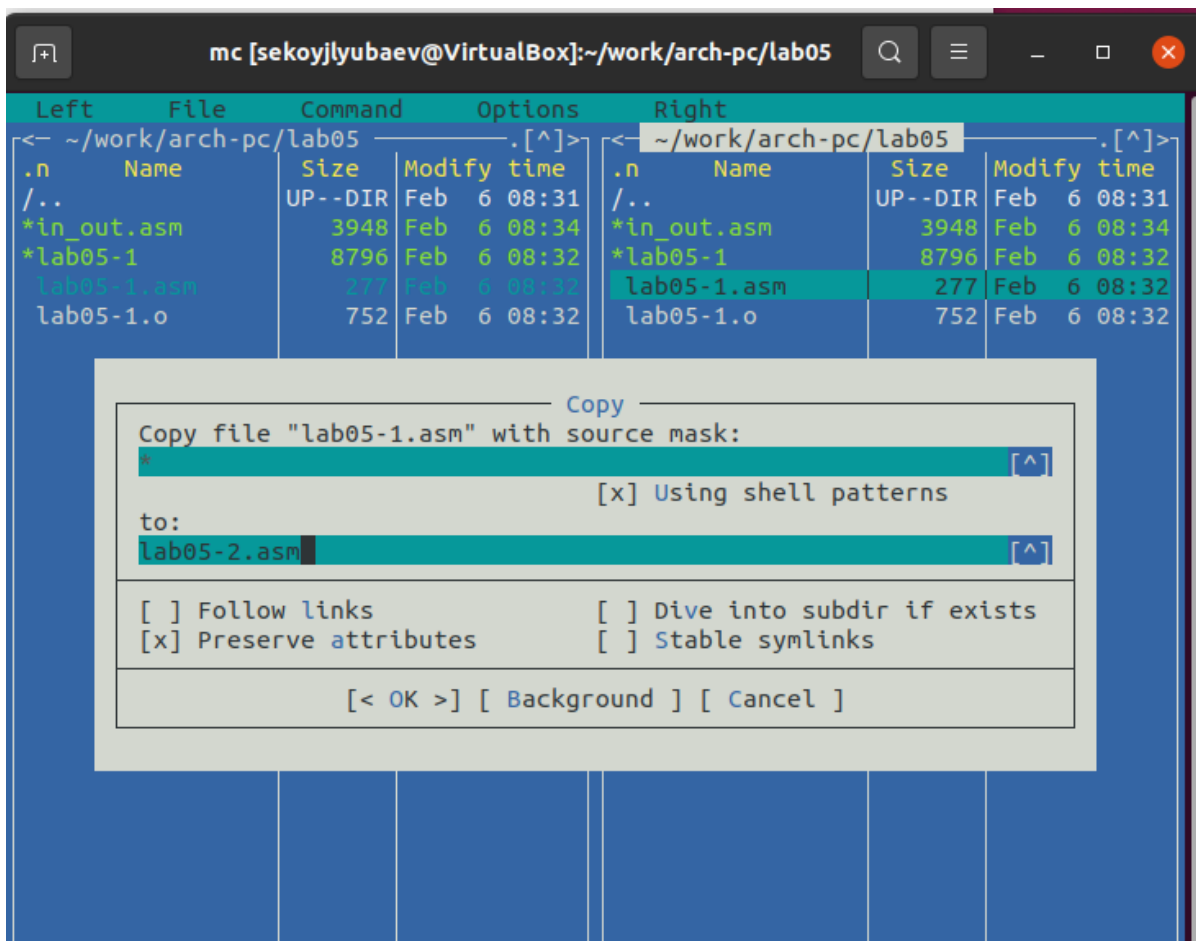
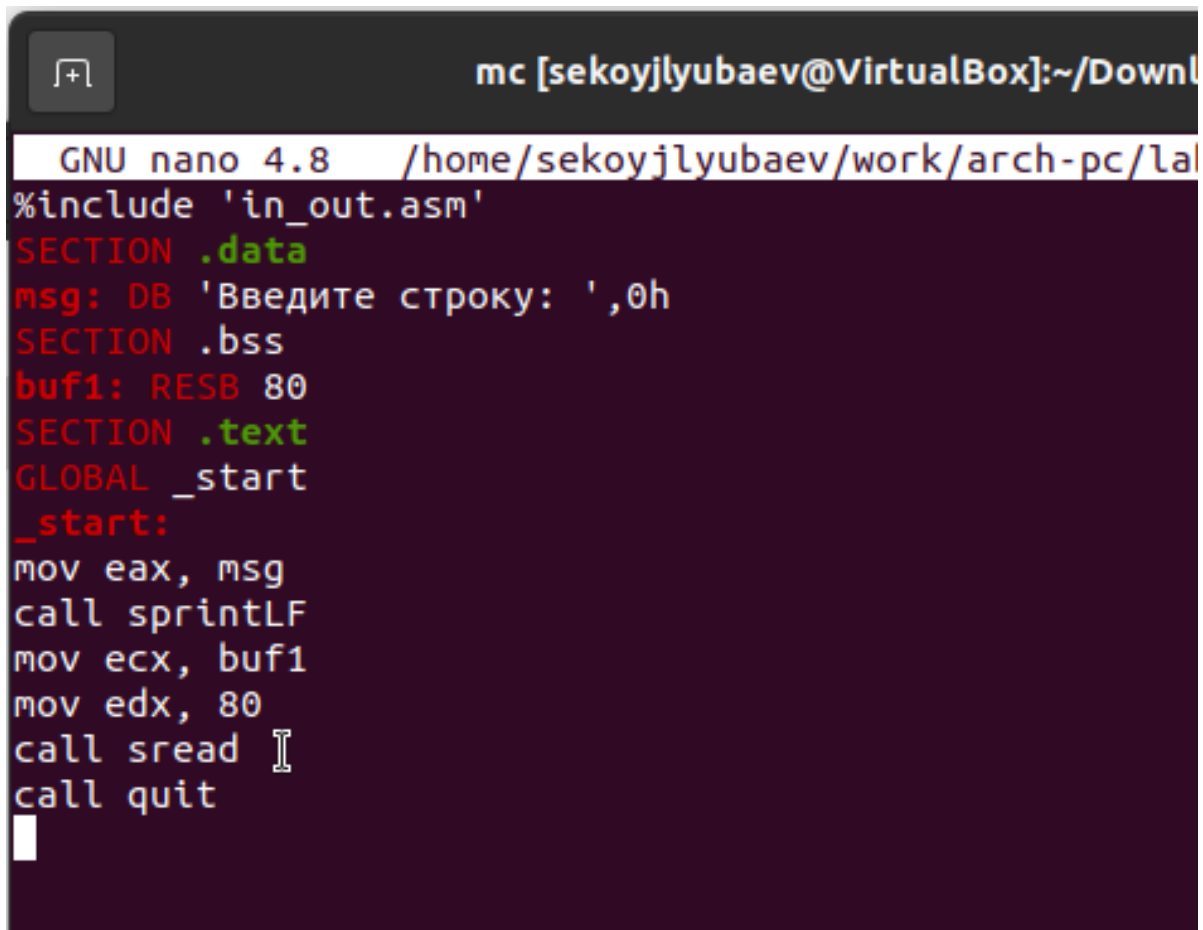


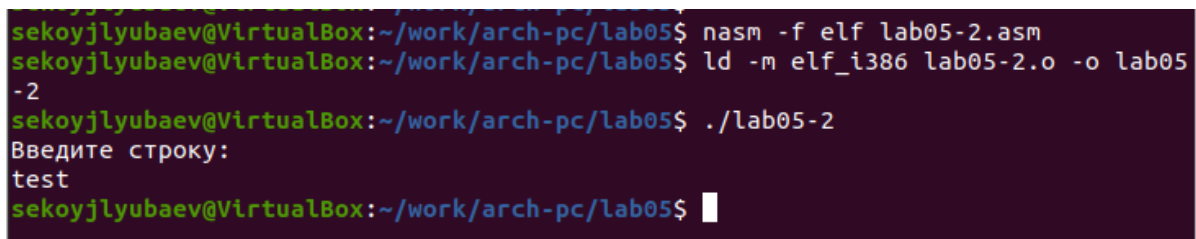
Рис. 2.8: Копирование файла lab05-1.asm

После этого я дописал код в файле lab05-2.asm, включив использование под-программ из файла in\_out.asm, скомпилировал и запустил программу для проверки.



```
mc [sekoyjlyubaev@VirtualBox]:~/Downl
GNU nano 4.8 /home/sekoyjlyubaev/work/arch-pc/lab05-2.asm
#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
```

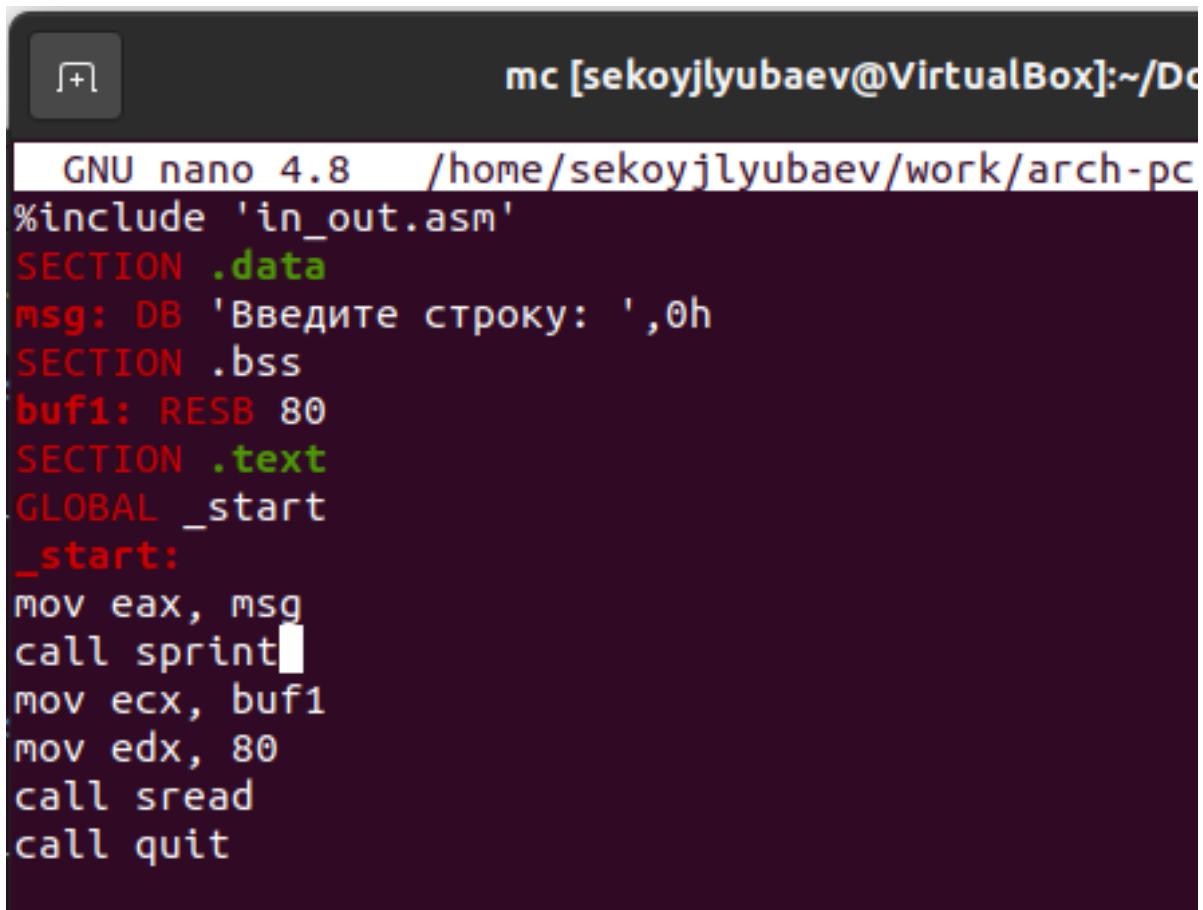
Рис. 2.9: Код программы lab05-2.asm



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

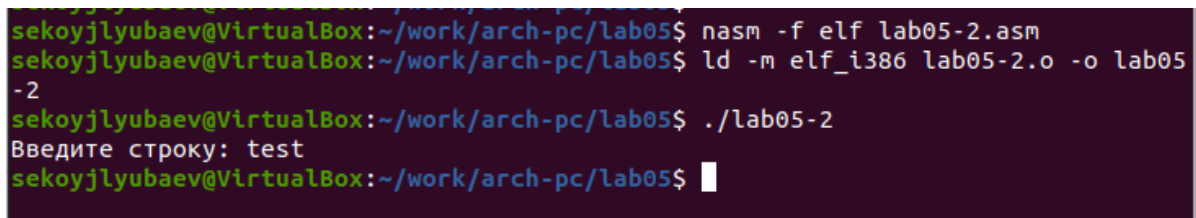
Рис. 2.10: Компиляция и запуск программы lab05-2.asm

В файле lab05-2.asm я заменил подпрограмму sprintLF на sprint, что после перекompиляции привело к отсутствию перевода строки после вывода текста.



```
mc [sekoyjlyubaev@VirtualBox]:~/Do
GNU nano 4.8 /home/sekoyjlyubaev/work/arch-pc
%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: Код программы lab05-2.asm



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

Рис. 2.12: Компиляция и запуск программы lab05-2.asm

## 2.1 Задание для самостоятельной работы

Я скопировал исходный код из файла lab05-1.asm и модифицировал его таким образом, чтобы программа запрашивала ввод строки с клавиатуры, затем

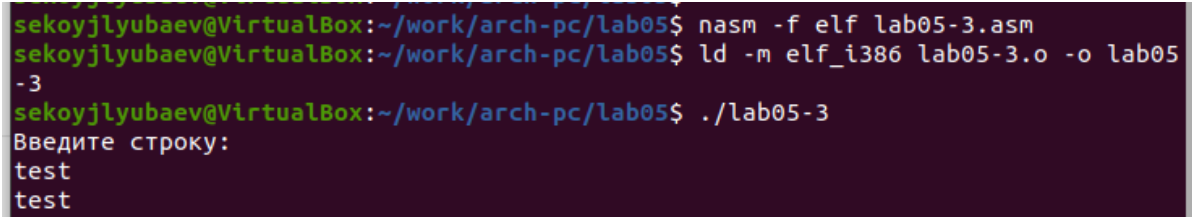
отображала эту строку на экране.



```
mc [sekoyjlyubaev@VirtualBox]:~/work/
GNU nano 4.8 /home/sekoyjlyubaev/work/arc
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
```

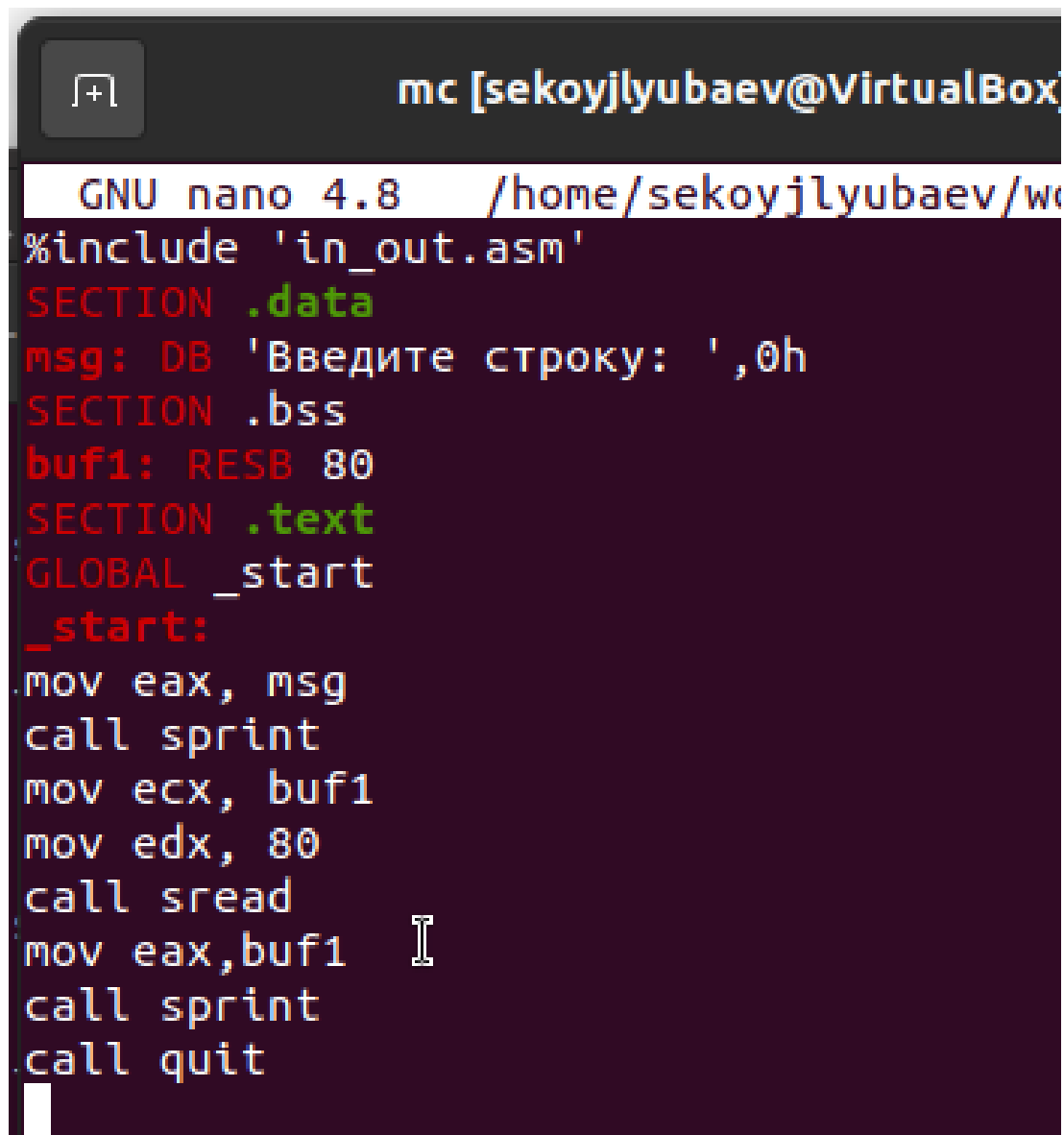
Рис. 2.13: Код программы lab05-3.asm

A terminal window with a dark background and light-colored text. The prompt is 'sekoyjlyubaev@VirtualBox:~/work/arch-pc/lab05\$'. The first command is 'nasm -f elf lab05-3.asm'. The second command is 'ld -m elf\_i386 lab05-3.o -o lab05-3'. The third command is './lab05-3'. The output shows 'Введите строку:' followed by 'test' on two separate lines.

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

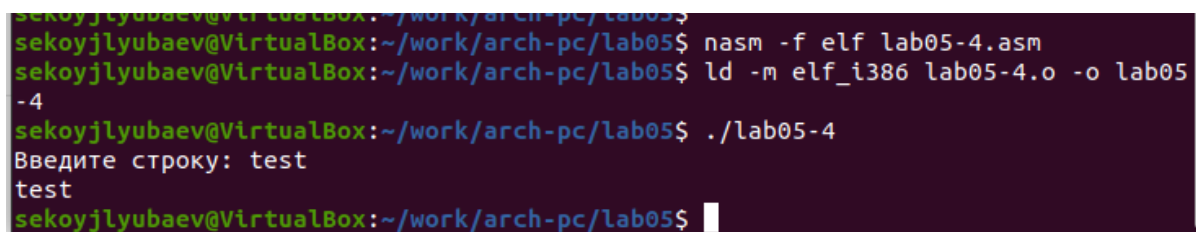
Рис. 2.14: Компиляция и запуск программы lab05-3.asm

Также я скопировал код из файла lab05-2.asm и внес в него изменения, но на этот раз я использовал подпрограммы из файла in\_out.asm для реализации алгоритма.



```
mc [sekoyjlyubaev@VirtualBox]
GNU nano 4.8 /home/sekoyjlyubaev/work
#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: Код программы lab05-4.asm



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

Рис. 2.16: Компиляция и запуск программы lab05-4.asm



## 3 Выводы

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