

# **Отчёт по лабораторной работе 5**

**Архитектура компьютеров и операционные системы**

Тарасова Алина НКАбд 05-23

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
4.1	Знакомство с Midnight Commander . . . . .	8
4.2	Подключение внешнего файла in_out.asm . . . . .	14
4.3	Задание для самостоятельной работы . . . . .	19
<b>5</b>	<b>Выводы</b>	<b>23</b>

## Список иллюстраций

4.1	Запуск Midnight Commander . . . . .	8
4.2	Создание каталога . . . . .	9
4.3	Создание файла lab05-1.asm . . . . .	10
4.4	Выбор редактора . . . . .	11
4.5	Программа lab05-1.asm . . . . .	12
4.6	Просмотр файла lab05-1.asm . . . . .	13
4.7	Запуск программы lab05-1.asm . . . . .	14
4.8	Копирование файла in_out.asm . . . . .	15
4.9	Копирование файла lab05-1.asm . . . . .	16
4.10	Программа lab05-2.asm . . . . .	17
4.11	Запуск программы lab05-2.asm . . . . .	17
4.12	Программа в файле lab05-2.asm . . . . .	18
4.13	Запуск программы lab05-2.asm . . . . .	18
4.14	Программа lab05-3.asm . . . . .	20
4.15	Запуск программы lab05-3.asm . . . . .	20
4.16	Программа lab05-4.asm . . . . .	21
4.17	Запуск программы lab05-4.asm . . . . .	22

## Список таблиц

# 1 Цель работы

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

## 2 Задание

1. Установить Midnight Commander
2. Изучить структуру программ
3. Изучить файл in\_out.asm
4. Дополнить программы по заданию.

### 3 Теоретическое введение

Midnight Commander (или просто mc) — это программа, которая позволяет просматривать структуру каталогов и выполнять основные операции по управлению файловой системой, т.е. mc является файловым менеджером. Midnight Commander позволяет сделать работу с файлами более удобной и наглядной.

Программа на языке ассемблера NASM, как правило, состоит из трёх секций: секция кода программы (SECTION .text), секция инициированных (известных во время компиляции) данных (SECTION .data) и секция неинициализированных данных (тех, под которые во время компиляции только отводится память, а значение присваивается в ходе выполнения программы) (SECTION .bss).

## 4 Выполнение лабораторной работы

### 4.1 Знакомство с Midnight Commander

Открываю Midnight Commander (рис. [4.1]), с помощью клавишь со стрелками и Enter перехожу в каталог ~/work/arch-pc. Далее нажимаю F7 и создаю каталог lab05 (рис. [4.2]).

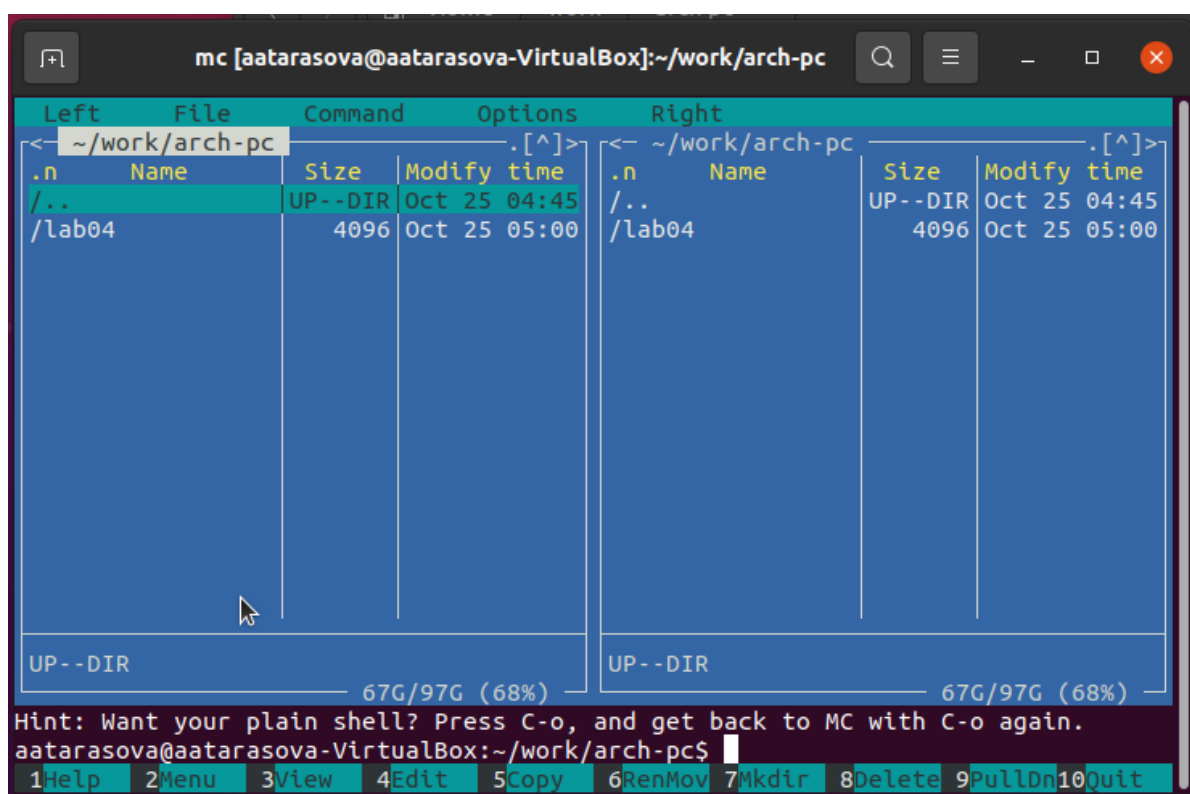


Рис. 4.1: Запуск Midnight Commander



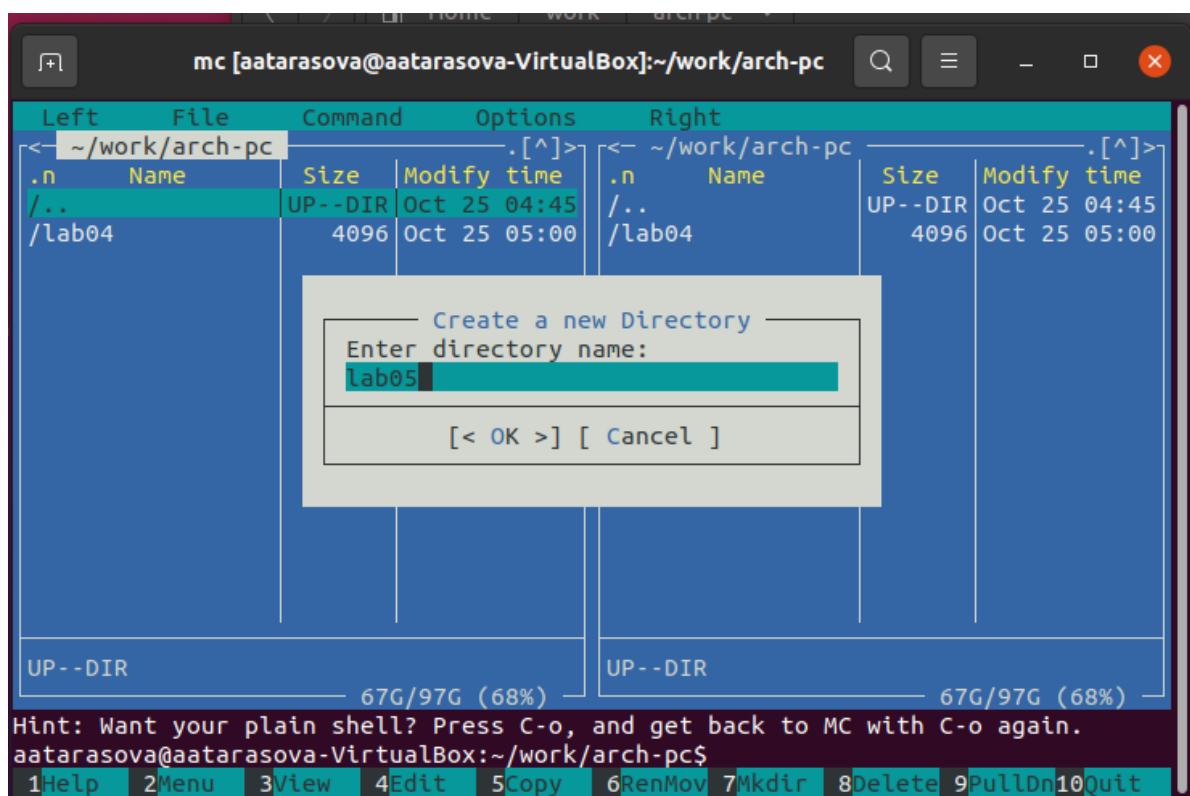


Рис. 4.2: Создание каталога

При помощи touch создаю файл lab05-1.asm (рис. [4.3])

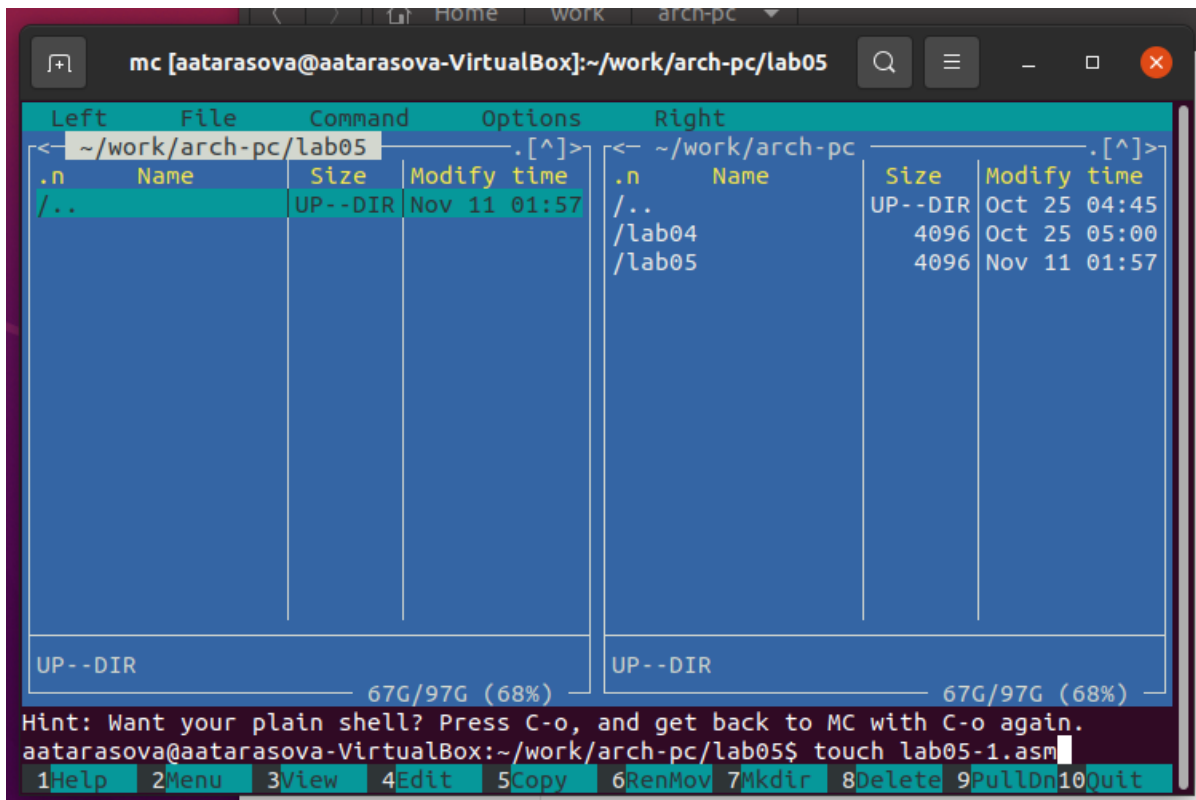
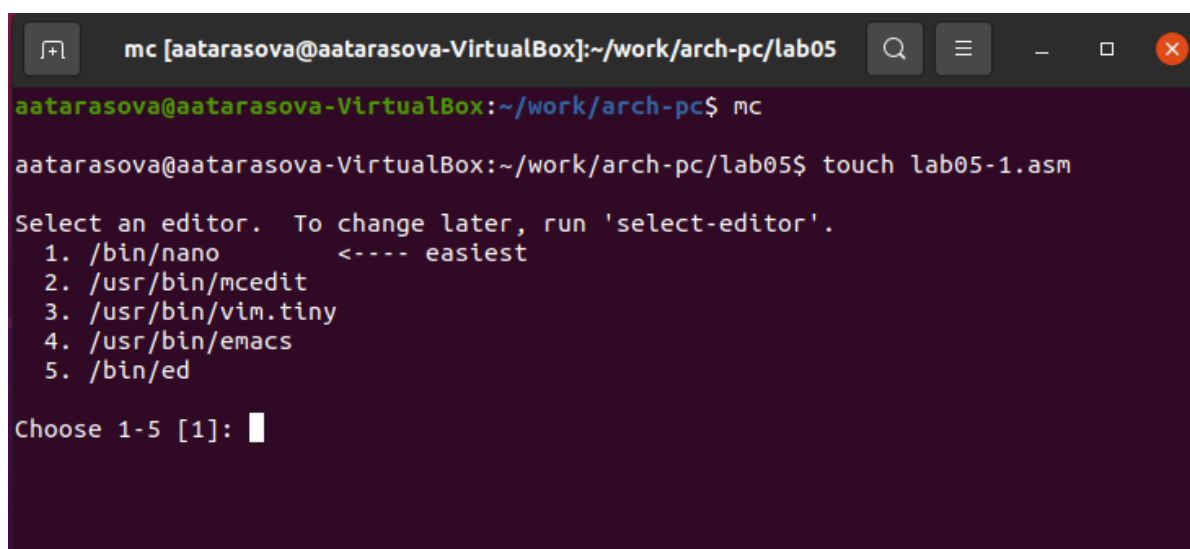


Рис. 4.3: Создание файла lab05-1.asm

Открываю файл на редактирование клавишей F4, выбираю редактор mcedit (рис. [4.4]), пишу код программы из задания. (рис. [4.5])

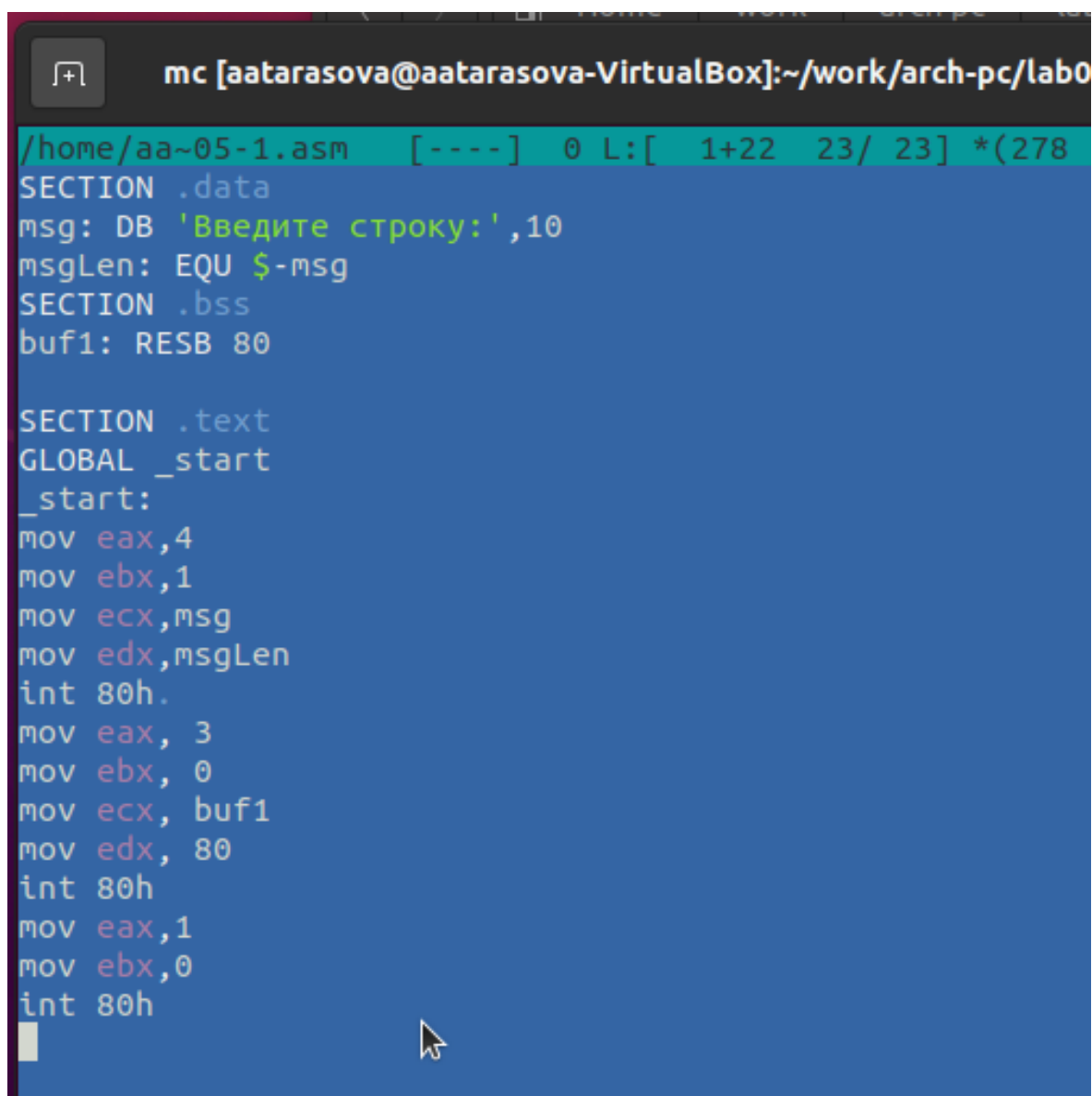


```
mc [aatarasova@aatarasova-VirtualBox]:~/work/arch-pc/lab05
aatarasova@aatarasova-VirtualBox:~/work/arch-pc$ mc
aatarasova@aatarasova-VirtualBox:~/work/arch-pc/lab05$ touch lab05-1.asm

Select an editor. To change later, run 'select-editor'.
 1. /bin/nano          <---- easiest
 2. /usr/bin/mcedit
 3. /usr/bin/vim.tiny
 4. /usr/bin/emacs
 5. /bin/ed

Choose 1-5 [1]:
```

Рис. 4.4: Выбор редактора

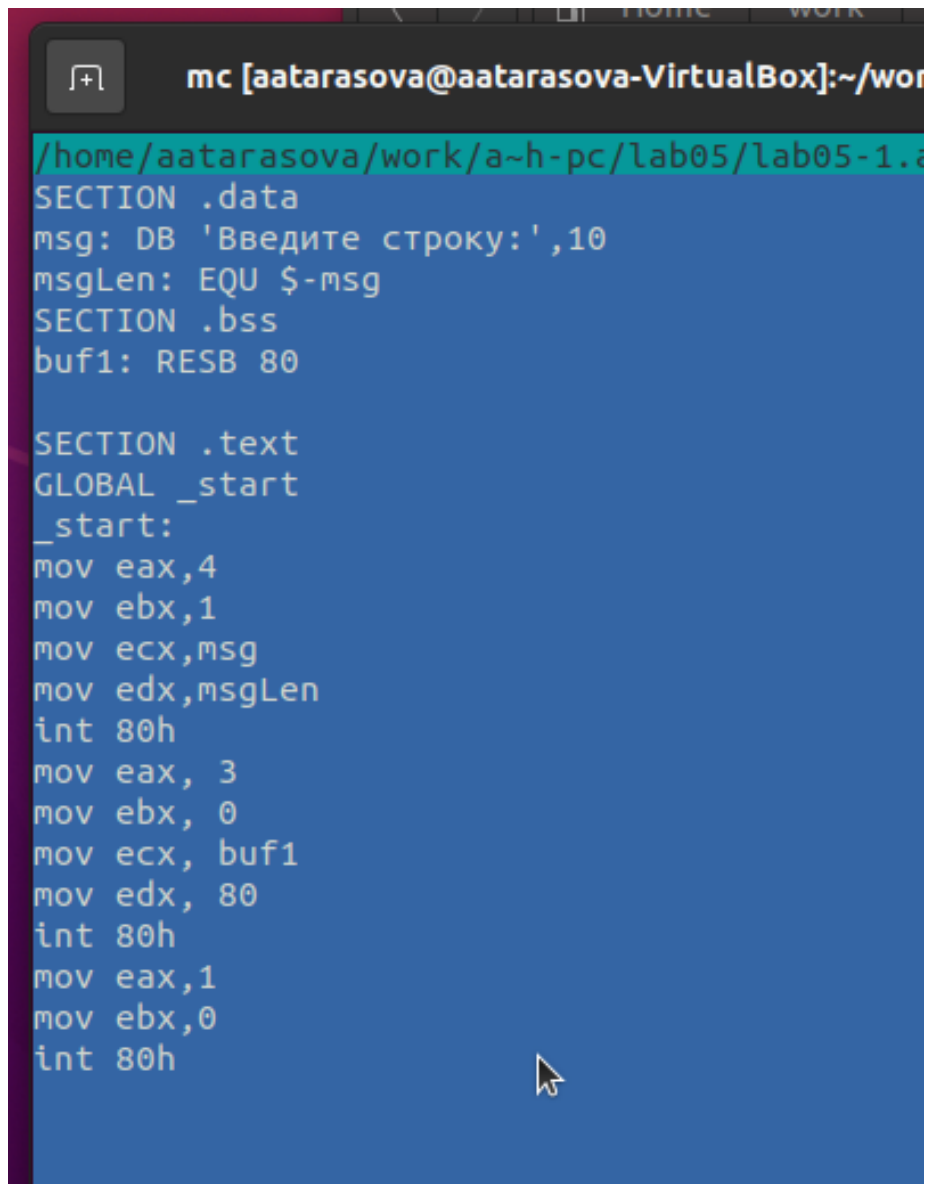


```
mc [aatarasova@aatarasova-VirtualBox]:~/work/arch-pc/lab0
/home/aa~05-1.asm [ - - - ] 0 L: [ 1+22 23/ 23 ] *(278
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

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. 4.5: Программа lab05-1.asm

Открываю файл на просмотр клавишей F3 и проверяю, что он содержит набранный код. (рис. [4.6])

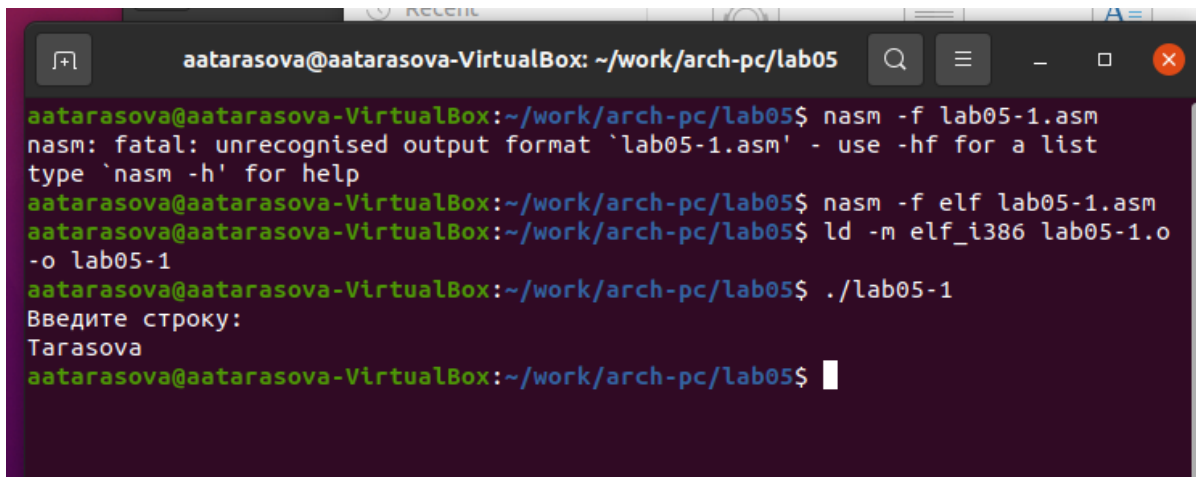


```
mc [aatarasova@aatarasova-VirtualBox]:~/work
/home/aatarasova/work/a~h-pc/lab05/lab05-1.a
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

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. 4.6: Просмотр файла lab05-1.asm

Транслирую файл программы в объектный файл, выполняю компоновку объектного файла, получился исполняемый файл программы. (рис. [4.7])



```
aatarasova@aatarasova-VirtualBox: ~/work/arch-pc/lab05
aatarasova@aatarasova-VirtualBox:~/work/arch-pc/lab05$ nasm -f lab05-1.asm
nasm: fatal: unrecognised output format `lab05-1.asm' - use -hf for a list
type `nasm -h' for help
aatarasova@aatarasova-VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab05-1.asm
aatarasova@aatarasova-VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 lab05-1.o
-o lab05-1
aatarasova@aatarasova-VirtualBox:~/work/arch-pc/lab05$ ./lab05-1
Введите строку:
Tarasova
aatarasova@aatarasova-VirtualBox:~/work/arch-pc/lab05$
```

Рис. 4.7: Запуск программы lab05-1.asm

## 4.2 Подключение внешнего файла in\_out.asm

Скачиваю файл in\_out.asm и размещаю его в рабочем каталоге. (рис. [4.8]) Для копирования используется клавиша F5. Для перемещения используется клавиша F6.

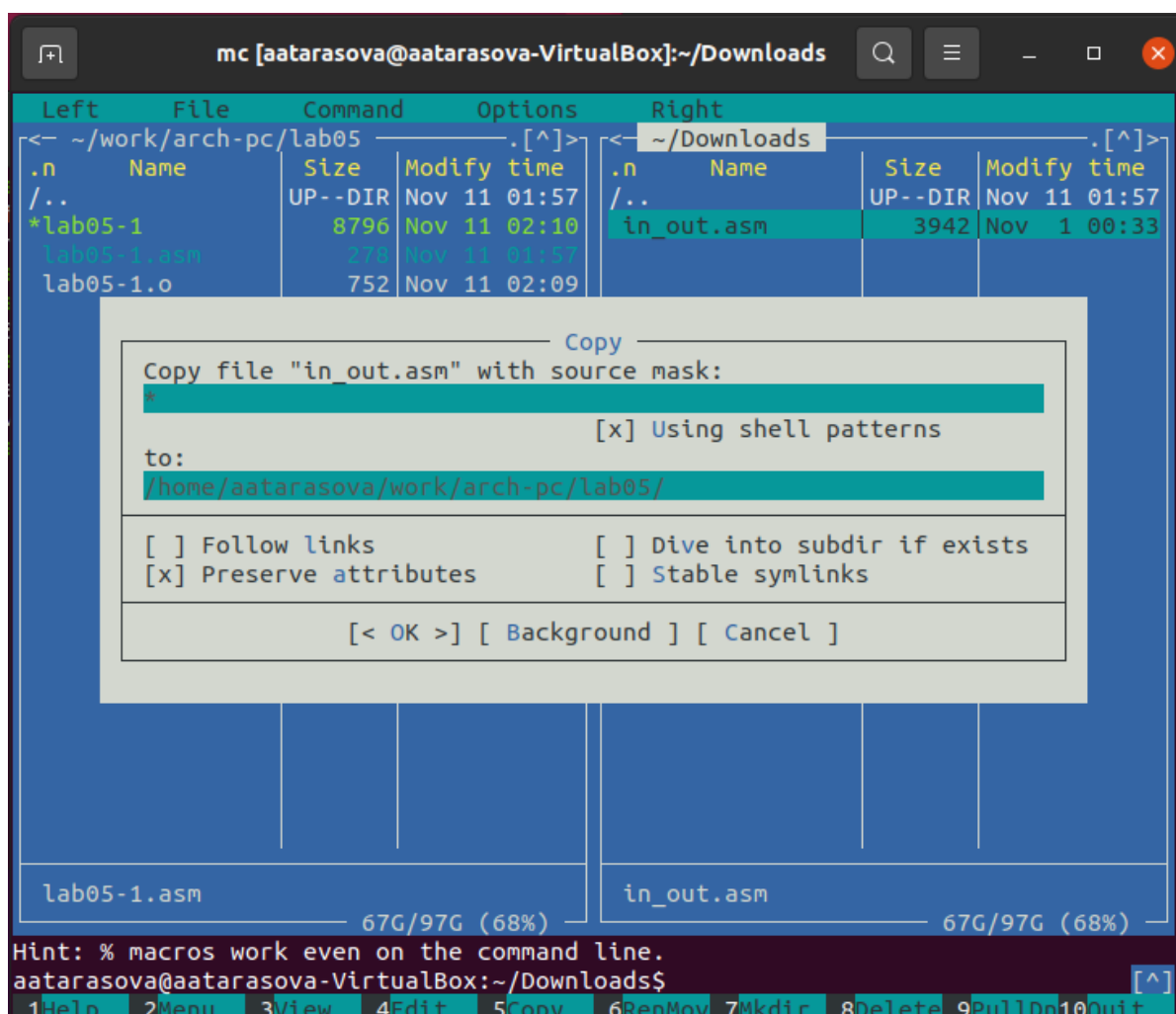


Рис. 4.8: Копирование файла in\_out.asm

Скопировала lab05-1.asm в lab05-2.asm. (рис. [4.9])

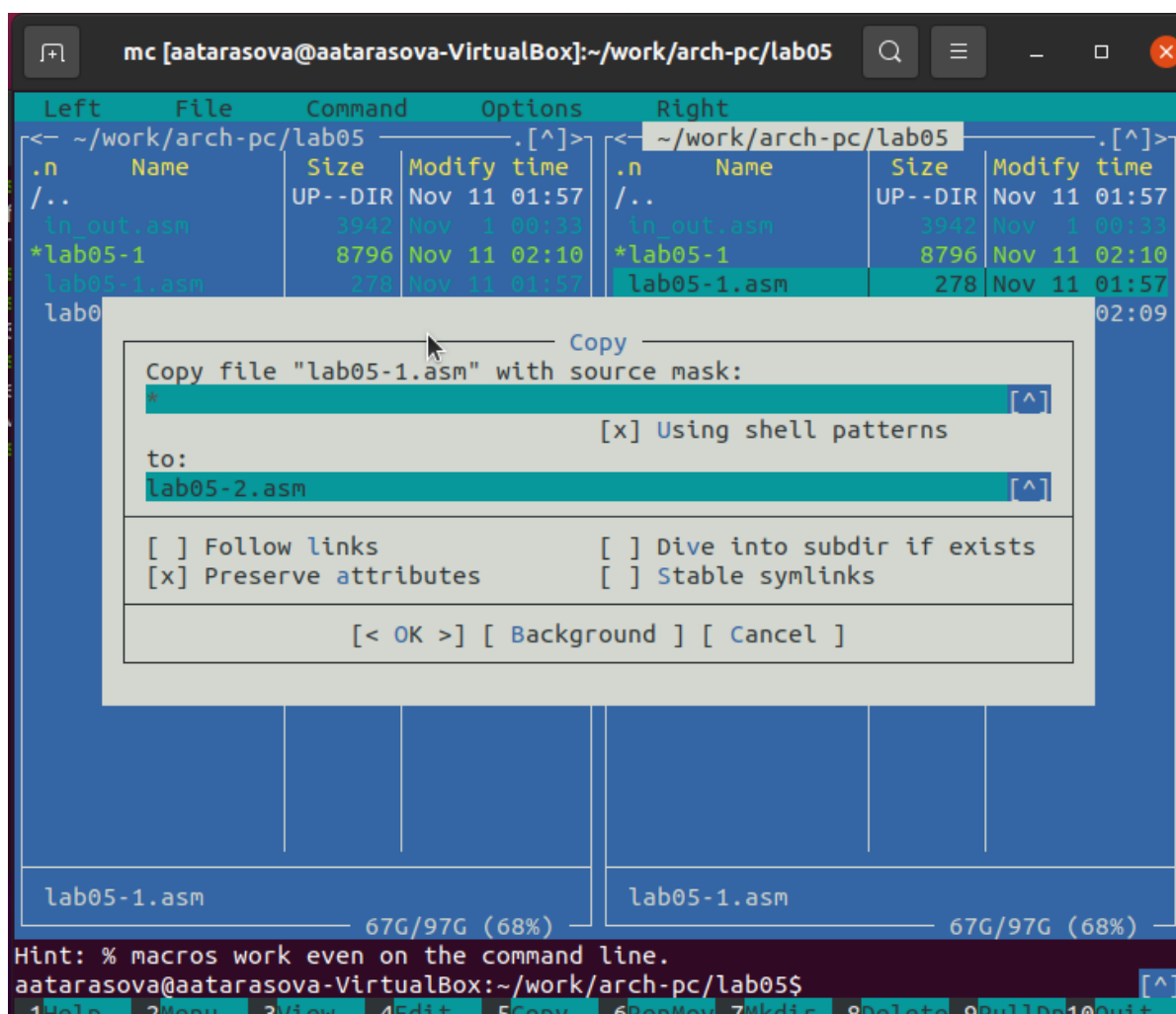
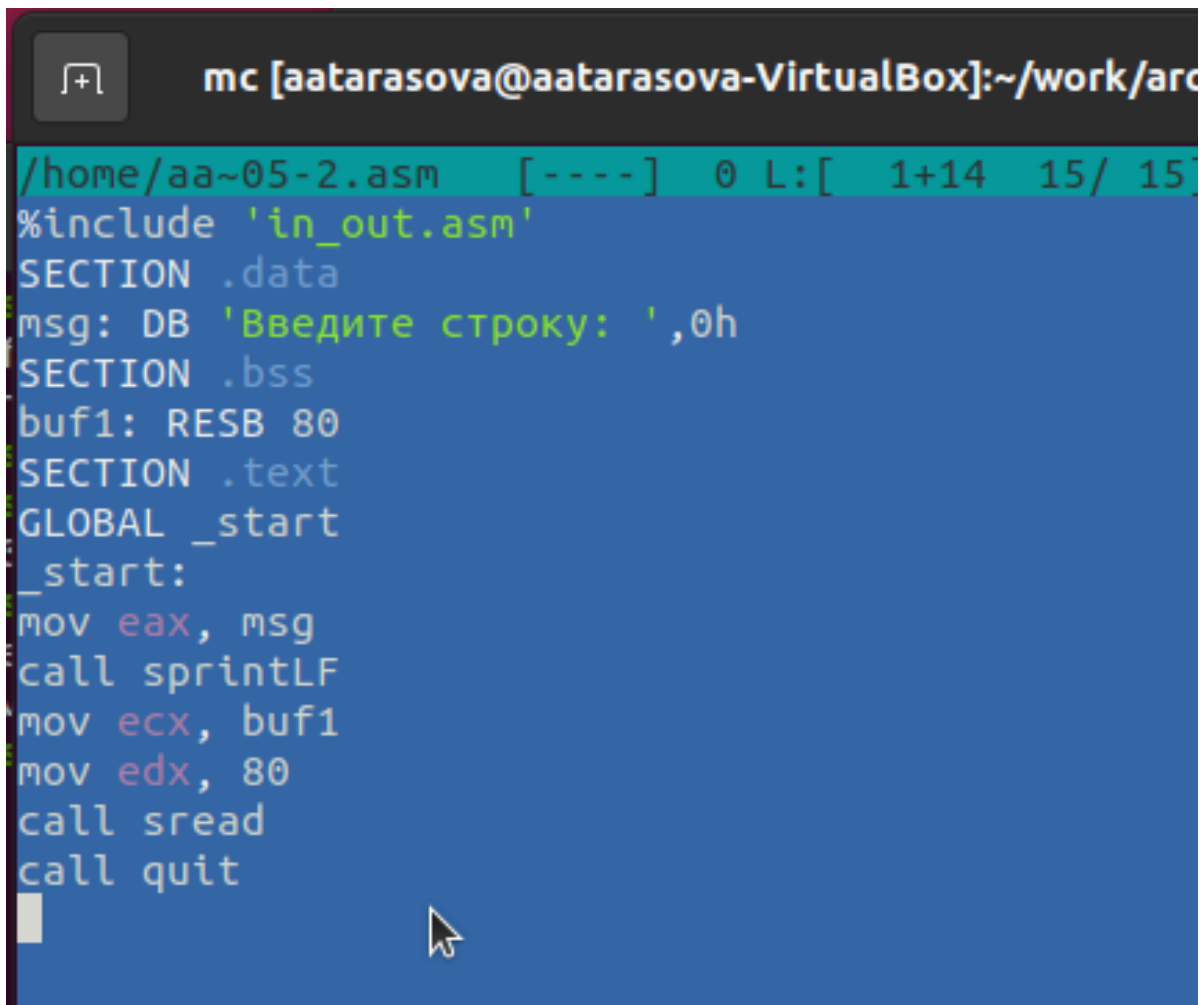


Рис. 4.9: Копирование файла lab05-1.asm

Пишу код программы lab05-2.asm с использованием подпрограмм из внешнего файла in\_out.asm. (рис. [4.10])

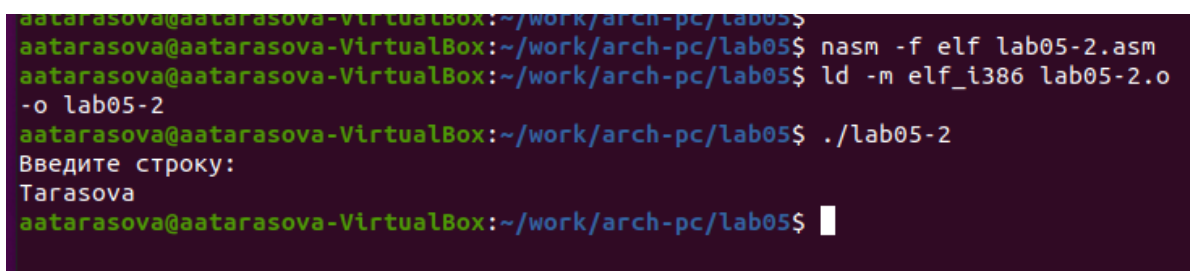




```
mc [aatarasova@aatarasova-VirtualBox]:~/work/arch-  
/home/aa~05-2.asm [----] 0 L:[ 1+14 15/ 15]  
%include 'in_out.asm'  
SECTION .data  
msg: DB 'Введите строку: ',0h  
SECTION .bss  
buf1: RESB 80  
SECTION .text  
GLOBAL _start  
_start:  
mov eax, msg  
call sprintf  
mov ecx, buf1  
mov edx, 80  
call sread  
call quit
```

Рис. 4.10: Программа lab05-2.asm

Скомпилирую программу и проверю запуск. (рис. [4.11])

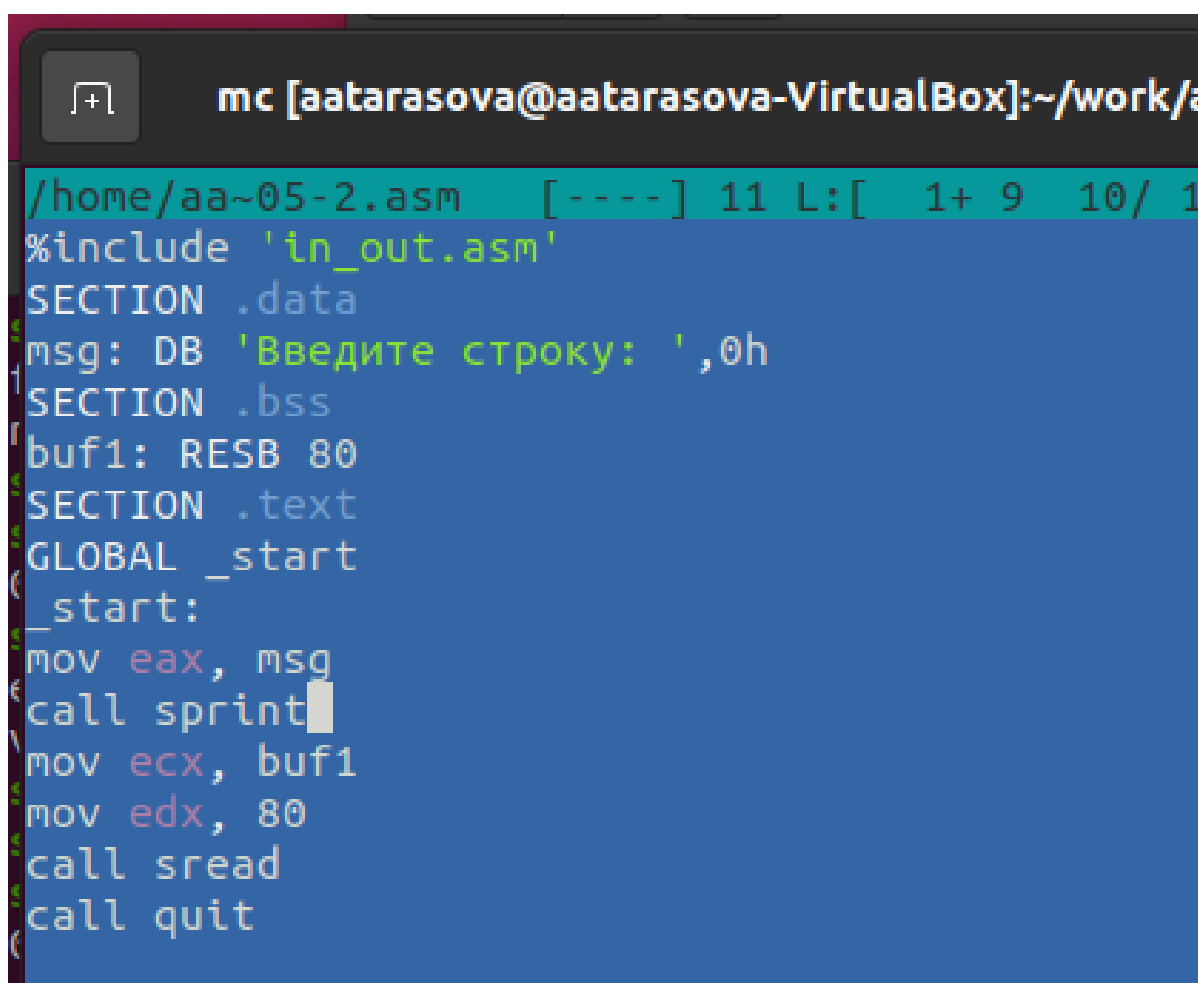


```
aatarasova@aatarasova-VirtualBox:~/work/arch-pc/lab05$  
aatarasova@aatarasova-VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab05-2.asm  
aatarasova@aatarasova-VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 lab05-2.o  
-o lab05-2  
aatarasova@aatarasova-VirtualBox:~/work/arch-pc/lab05$ ./lab05-2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

Рис. 4.11: Запуск программы lab05-2.asm

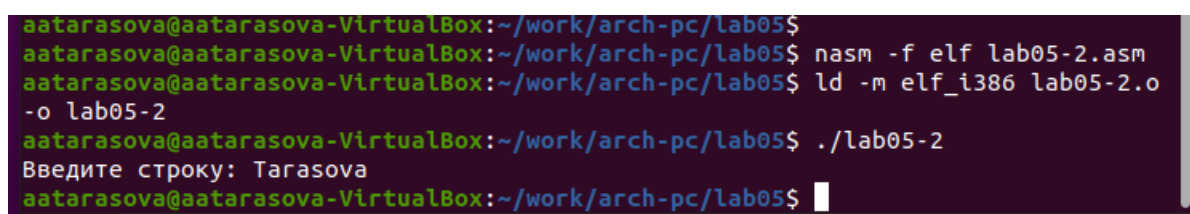
В файле lab5-2.asm заменила подпрограмму sprintf на sprint. Заново собрала

исполняемый файл. (рис. [4.12]) (рис. [4.13])



```
mc [aatarasova@aatarasova-VirtualBox]:~/work/a
/home/aa~05-2.asm [ - - - - ] 11 L: [ 1+ 9 10/ 1
%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. 4.12: Программа в файле lab05-2.asm



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

Рис. 4.13: Запуск программы lab05-2.asm

Теперь после вывода строки она не завершается символом перехода на новую строку.

### 4.3 Задание для самостоятельной работы

Скопировала программу lab05-1.asm и изменила код, так чтобы она работала по следующему алгоритму: (рис. [4.14]) (рис. [4.15])

- вывести приглашение типа “Введите строку:”;
- ввести строку с клавиатуры;
- вывести введённую строку на экран.

```
mc [aatarasova@aatarasova-VirtualBox]:~/work/arch-pc
/home/aa~05-3.asm [----] 11 L:[ 1+17 18/ 28] *(
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

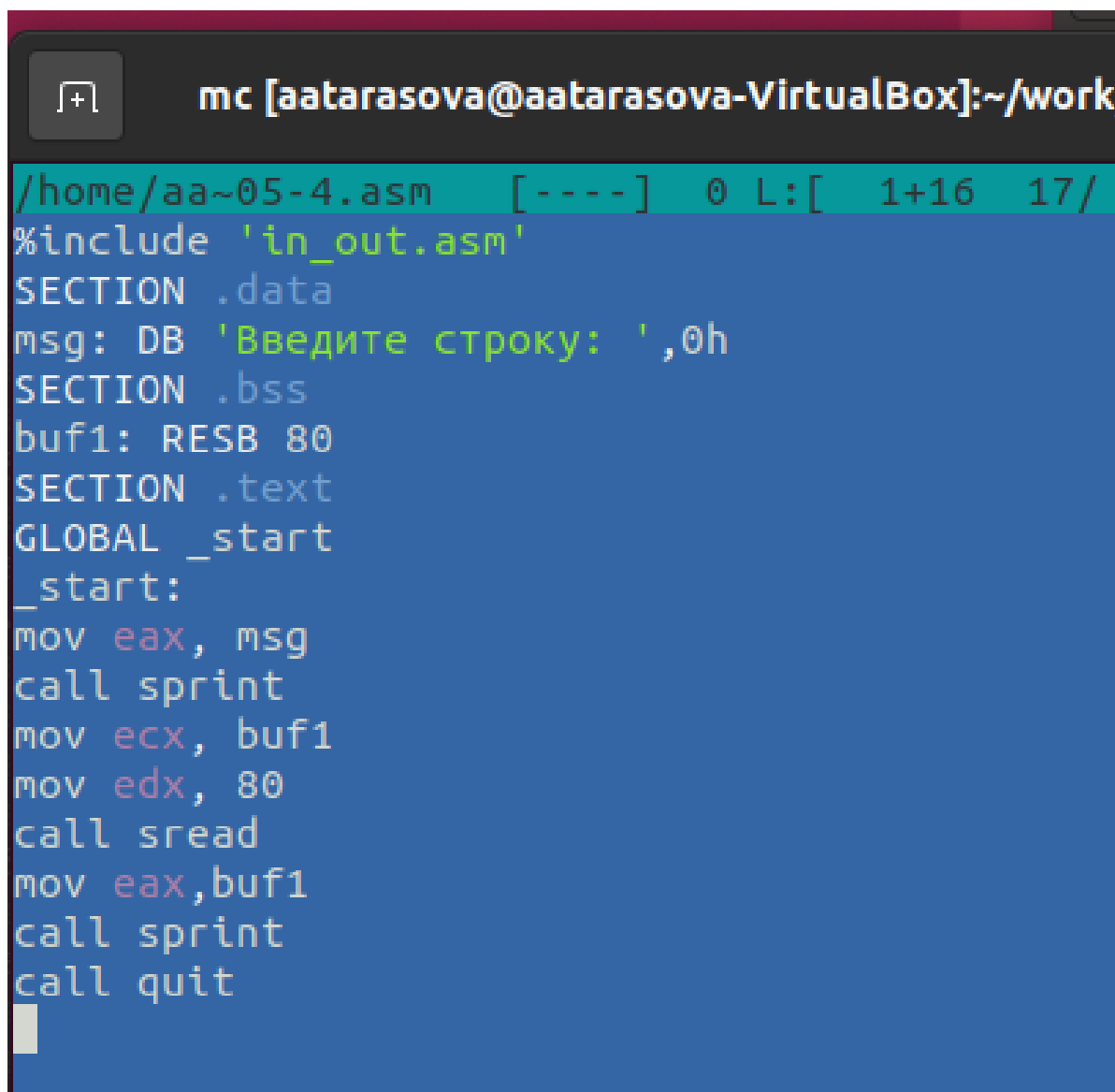
SECTION .text
GLOBAL _start
_start:
mov eax,4
mov ebx,1
mov ecx,msg
mov edx,msgLen
int 80h.
mov eax, 3
mov ebx, 0
mov ecx, buf1
mov edx, 80
int 80h.
mov eax,4
mov ebx,1
mov ecx,buf1
mov edx,80
int 80h
mov eax,1
mov ebx,0
int 80h
```

Рис. 4.14: Программа lab05-3.asm

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

Рис. 4.15: Запуск программы lab05-3.asm

Аналогично скопировала программу lab05-2.asm и изменила код, но теперь использовал подпрограммы из файла in\_out.asm. (рис. [4.16]) (рис. [4.17])



```
mc [aatarasova@aatarasova-VirtualBox]:~/work  
/home/aa~05-4.asm [----] 0 L:[ 1+16 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. 4.16: Программа lab05-4.asm

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

Рис. 4.17: Запуск программы lab05-4.asm

## 5 Выводы

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