

Отчёт по лабораторной работе 5

дисциплина: Архитектура компьютера

Элиана Сулейманова

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
2.1	Знакомство с Midnight Commander	6
2.2	Подключение внешнего файла in_out.asm	11
2.3	Задание для самостоятельной работы	15
3	Выводы	18

Список иллюстраций

2.1	Запуск Midnight Commander	7
2.2	Создание нового каталога	7
2.3	Создание файла lab05-1.asm	8
2.4	Написание программы lab05-1.asm	9
2.5	Просмотр кода lab05-1.asm	10
2.6	Тестовый запуск lab05-1.asm	10
2.7	Перемещение файла in_out.asm	11
2.8	Создание копии файла lab05-1.asm	12
2.9	Добавление подпрограмм в lab05-2.asm	13
2.10	Запуск программы lab05-2.asm	13
2.11	Обновлённая версия lab05-2.asm	14
2.12	Запуск обновлённой lab05-2.asm	14
2.13	Редактирование программы lab05-3.asm	15
2.14	Тестовый запуск lab05-3.asm	16
2.15	Обновлённая программа lab05-4.asm	16
2.16	Тестовый запуск lab05-4.asm	17

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 Знакомство с Midnight Commander

Запускаю Midnight Commander (см. рис. 2.1), используя клавиши со стрелками и Enter, перехожу в каталог `~/work/arch-rc`. Затем создаю новый каталог под названием `lab05`, нажав F7 (см. рис. 2.2).

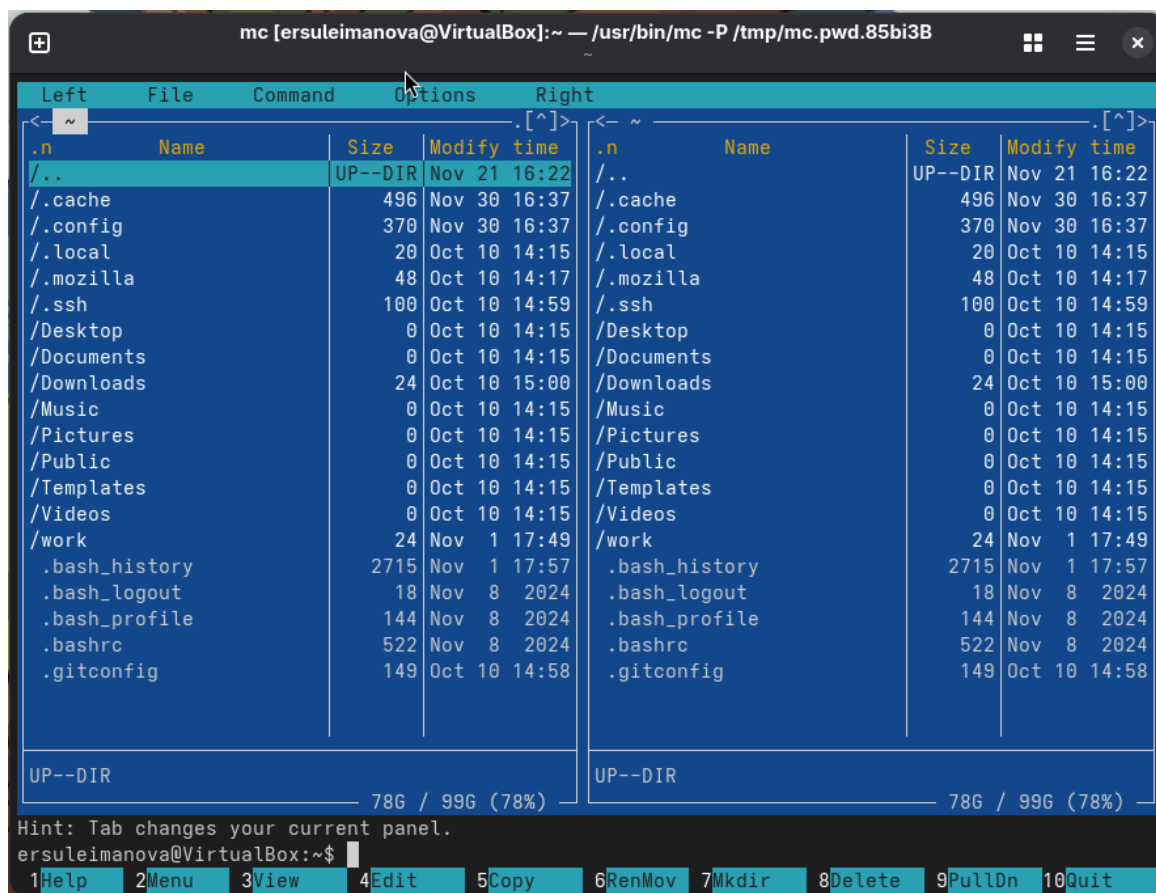


Рисунок 2.1: Запуск Midnight Commander

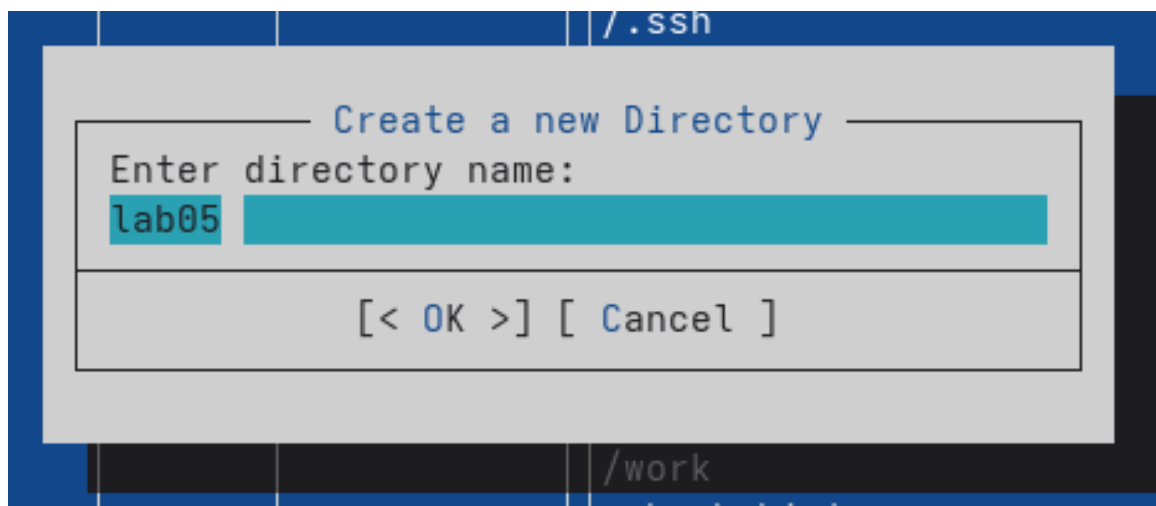


Рисунок 2.2: Создание нового каталога

Использую команду touch, чтобы создать файл lab05-1.asm (см. рис. 2.3).

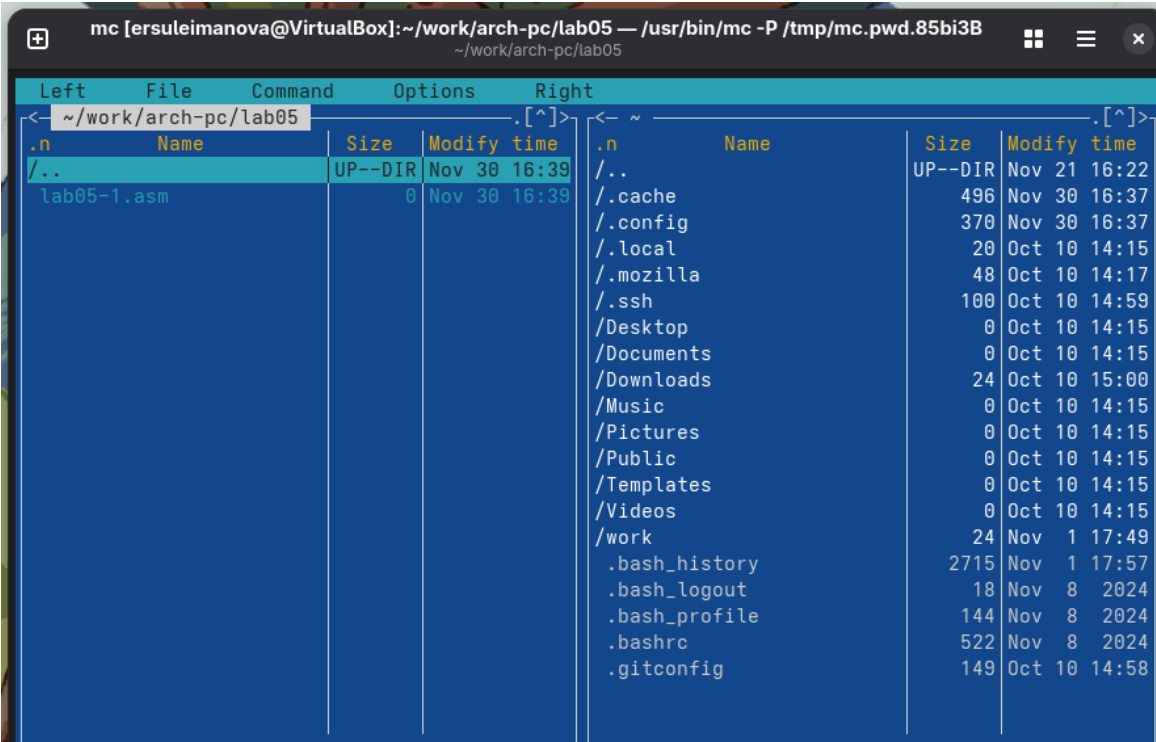


Рисунок 2.3: Создание файла lab05-1.asm

Открываю файл для редактирования с помощью F4, выбираю редактор mcedit, и пишу программу в соответствии с заданием (см. рис. 2.4).

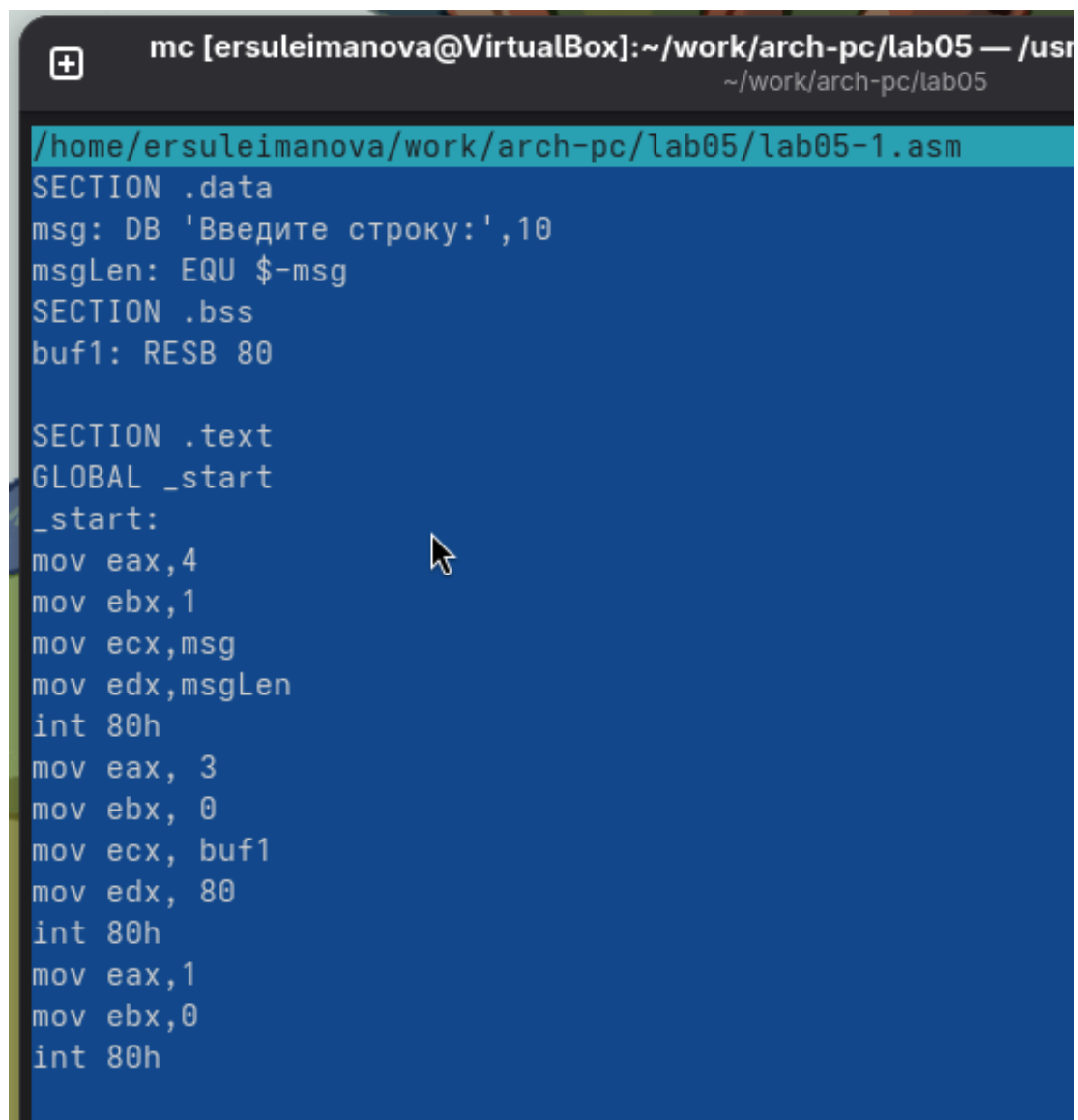


```
mc [ersuleimanova@VirtualBox]:~/work/arch
lab05-1.asm [----] 10 L:[ 1+15 16,
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

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 2.4: Написание программы lab05-1.asm

Для проверки содержимого файла просматриваю его с помощью F3 и убеждаюсь в корректности написанного кода (см. рис. 2.5).



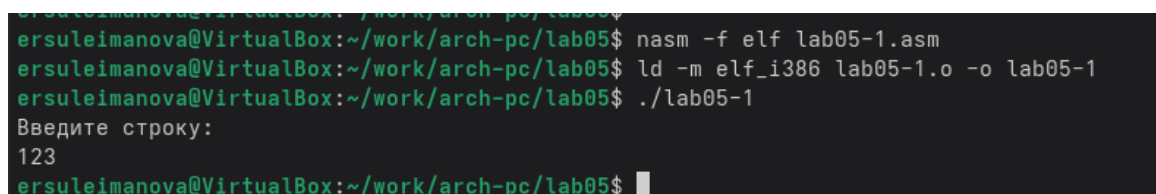
```
mc [ersuleimanova@VirtualBox]:~/work/arch-pc/lab05 — /usr
~/work/arch-pc/lab05

/home/ersuleimanova/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
```

Рисунок 2.5: Просмотр кода lab05-1.asm

Транслирую исходный код в объектный файл, затем выполняю компоновку для создания исполняемого файла программы (см. рис. 2.6).



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

Рисунок 2.6: Тестовый запуск lab05-1.asm

2.2 Подключение внешнего файла in_out.asm

Скачиваю файл `in_out.asm` и переношу его в рабочий каталог (см. рис. 2.7). Для копирования использую F5, для перемещения — F6.

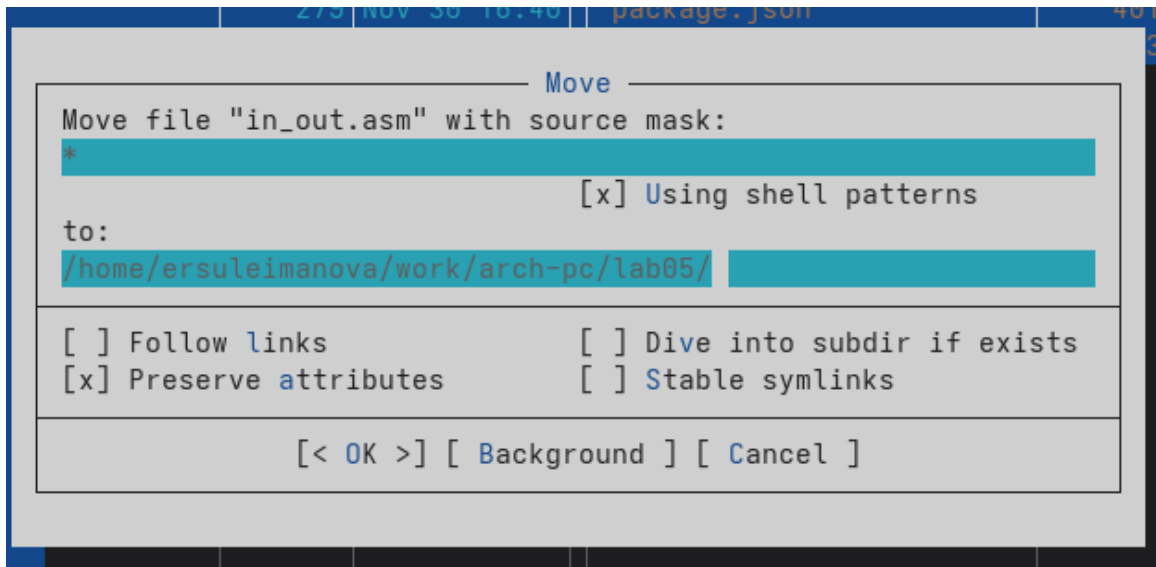


Рисунок 2.7: Перемещение файла `in_out.asm`

Копирую файл `lab05-1.asm` и создаю его копию под именем `lab05-2.asm` (см. рис. 2.8).

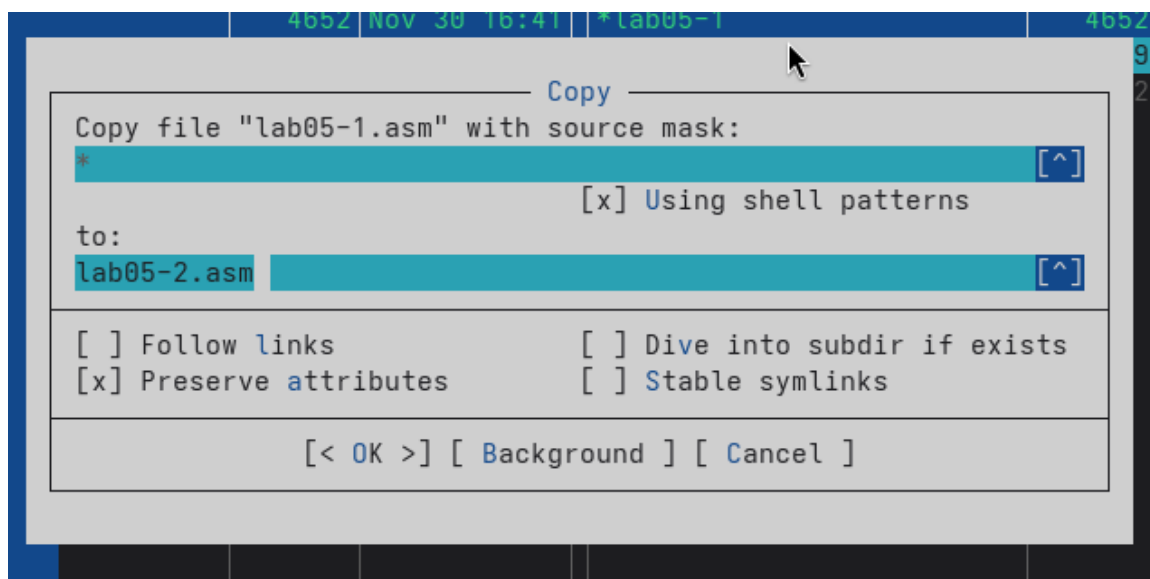


Рисунок 2.8: Создание копии файла lab05-1.asm

Редактирую код lab05-2.asm, добавляя подпрограммы из файла in_out.asm (см. рис. 2.9).

```
lab05-2.asm [----] 0 L:[ 1+14 15/ 15] *(2
#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
```

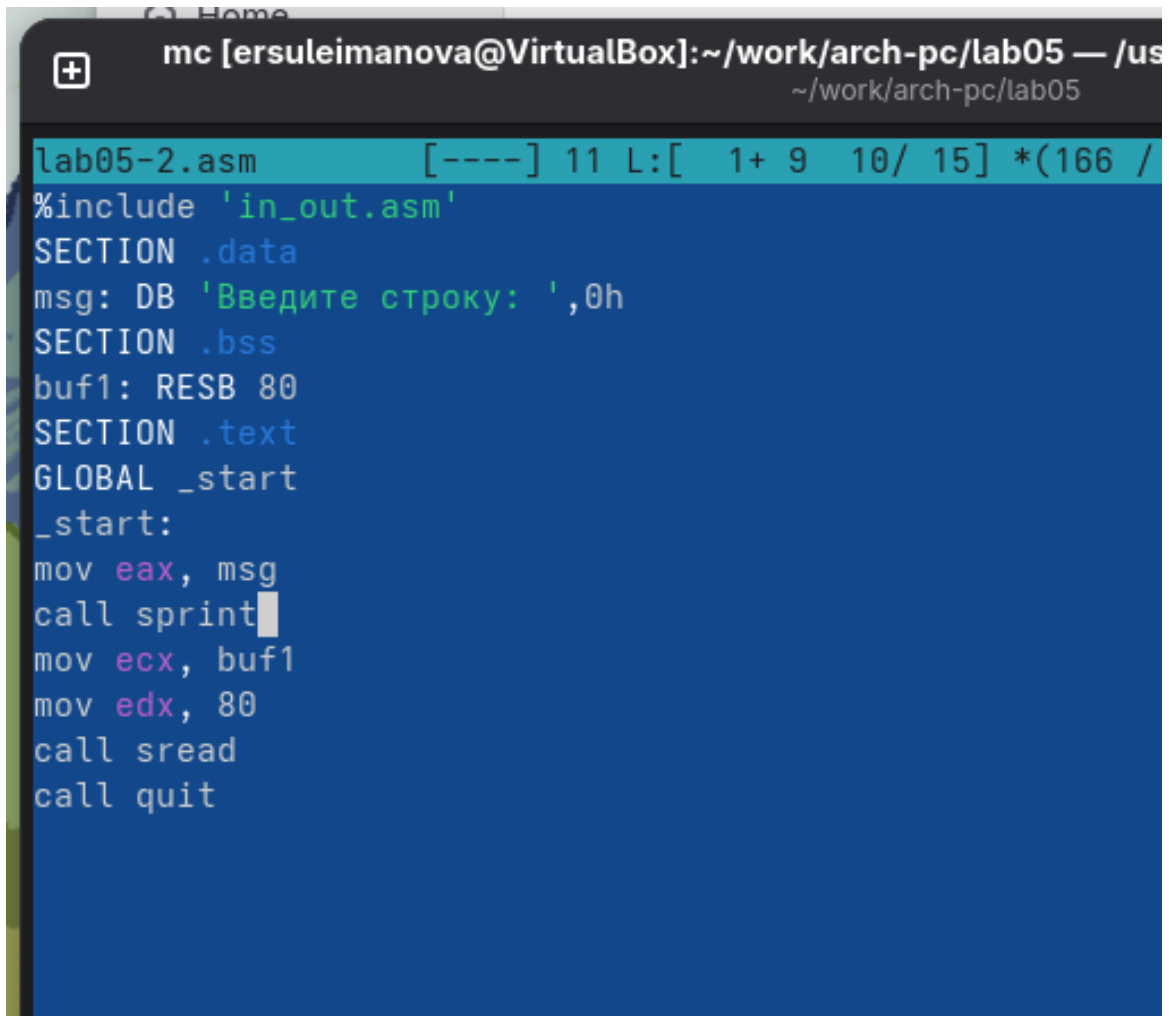
Рисунок 2.9: Добавление подпрограмм в lab05-2.asm

Компилирую программу и проверяю её выполнение (см. рис. 2.10).

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

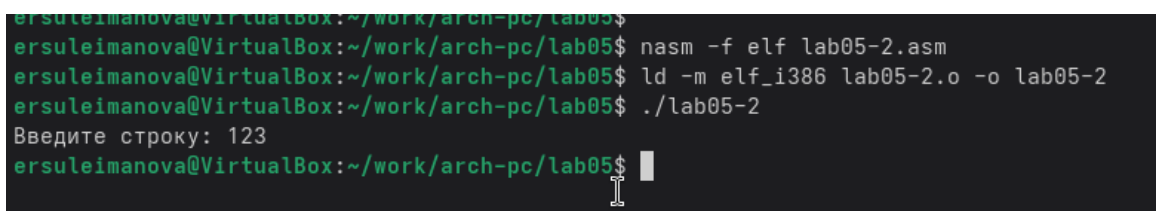
Рисунок 2.10: Запуск программы lab05-2.asm

В файле lab05-2.asm заменяю подпрограмму sprintf на sprintf, после чего повторно собираю программу (см. рис. 2.11 и 2.12).



```
mc [ersuleimanova@VirtualBox]:~/work/arch-pc/lab05 — /us
~/work/arch-pc/lab05
lab05-2.asm [----] 11 L:[ 1+ 9 10/ 15] *(166 /
#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 2.11: Обновлённая версия lab05-2.asm



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

Рисунок 2.12: Запуск обновлённой lab05-2.asm

Теперь программа выводит строку без переноса на новую строку.

2.3 Задание для самостоятельной работы

Создаю копию программы lab05-1.asm и изменяю код для выполнения следующего алгоритма (см. рис. 2.13 и 2.14): - отображает запрос «Введите строку:»; - принимает строку с клавиатуры; - выводит введенную строку на экран.



```
lab05-3.asm [----] 7 L:[ 1+23 24/ 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
```

Рисунок 2.13: Редактирование программы lab05-3.asm

```

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

```

Рисунок 2.14: Тестовый запуск lab05-3.asm

Аналогично, копирую lab05-2.asm и изменяю код, добавляя подпрограммы из файла in_out.asm (см. рис. 2.15 и 2.16).

```

lab05-4.asm      [----]  9 L:[  1+15  16/ 16] *(238
%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 2.15: Обновлённая программа lab05-4.asm


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

Рисунок 2.16: Тестовый запуск lab05-4.asm

3 Выводы

В ходе работы освоены базовые навыки написания ассемблерных программ, а также инструкции `mov` и `int`.