

# **Отчёт по лабораторной работе 5**

**дисциплина: Архитектура компьютера**

Кайнова Екатерина Андреевна НПИбд-03-24

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
2.1	Знакомство с Midnight Commander . . . . .	6
2.2	Подключение внешнего файла in_out.asm . . . . .	12
2.3	Задание для самостоятельной работы . . . . .	16
<b>3</b>	<b>Выводы</b>	<b>19</b>

## Список иллюстраций

2.1	Запуск Midnight Commander . . . . .	6
2.2	Создание нового каталога . . . . .	7
2.3	Создание файла lab05-1.asm . . . . .	8
2.4	редактор mcedit . . . . .	9
2.5	Написание программы lab05-1.asm . . . . .	10
2.6	Просмотр кода lab05-1.asm . . . . .	11
2.7	Тестовый запуск lab05-1.asm . . . . .	12
2.8	Перемещение файла in_out.asm . . . . .	12
2.9	Создание копии файла lab05-1.asm . . . . .	13
2.10	Добавление подпрограмм в lab05-2.asm . . . . .	14
2.11	Запуск программы lab05-2.asm . . . . .	14
2.12	Обновлённая версия lab05-2.asm . . . . .	15
2.13	Запуск обновлённой lab05-2.asm . . . . .	15
2.14	Редактирование программы lab05-3.asm . . . . .	17
2.15	Тестовый запуск lab05-3.asm . . . . .	17
2.16	Обновлённая программа lab05-4.asm . . . . .	18
2.17	Тестовый запуск lab05-4.asm . . . . .	18

## Список таблиц

# 1 Цель работы

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



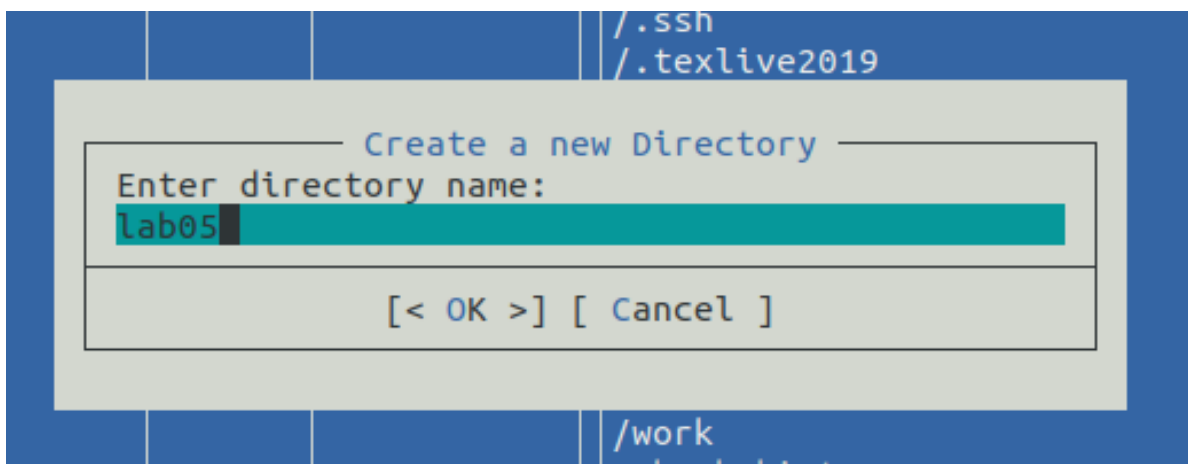


Рис. 2.2: Создание нового каталога

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

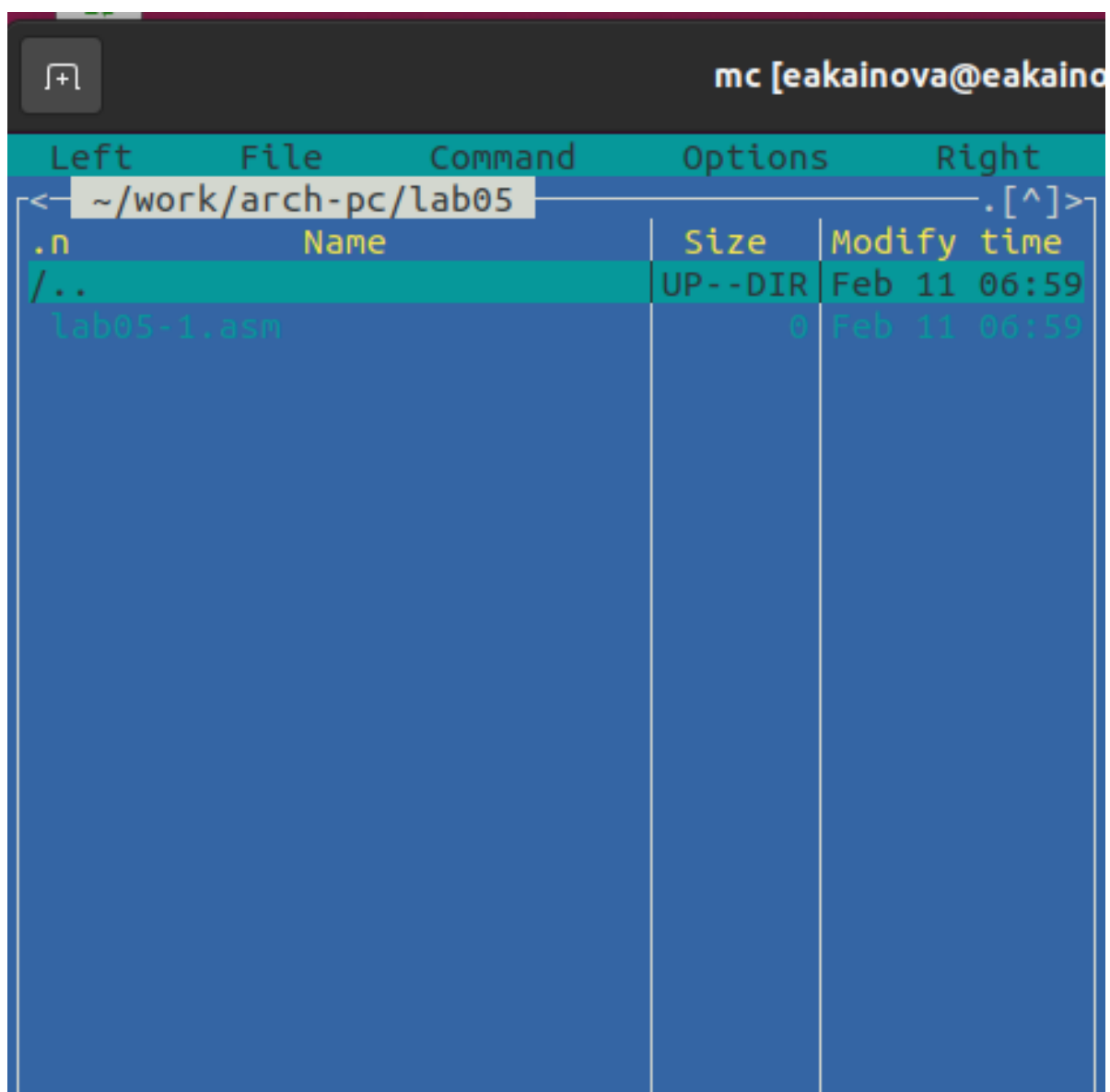
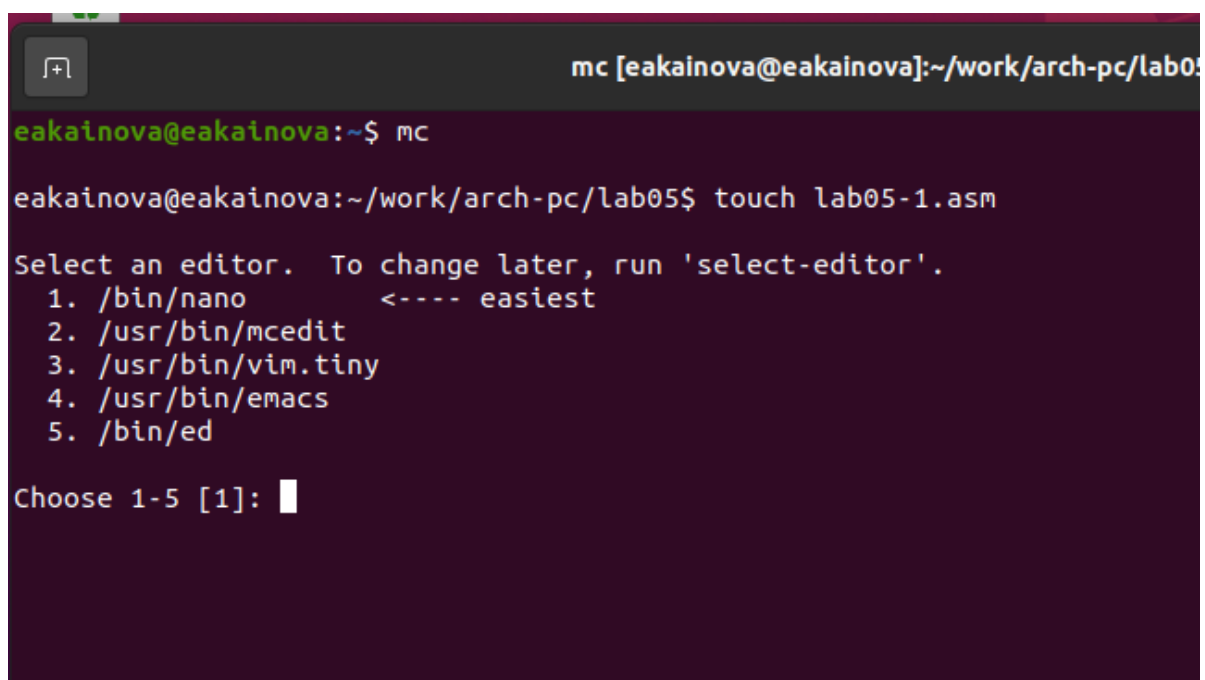


Рис. 2.3: Создание файла lab05-1.asm

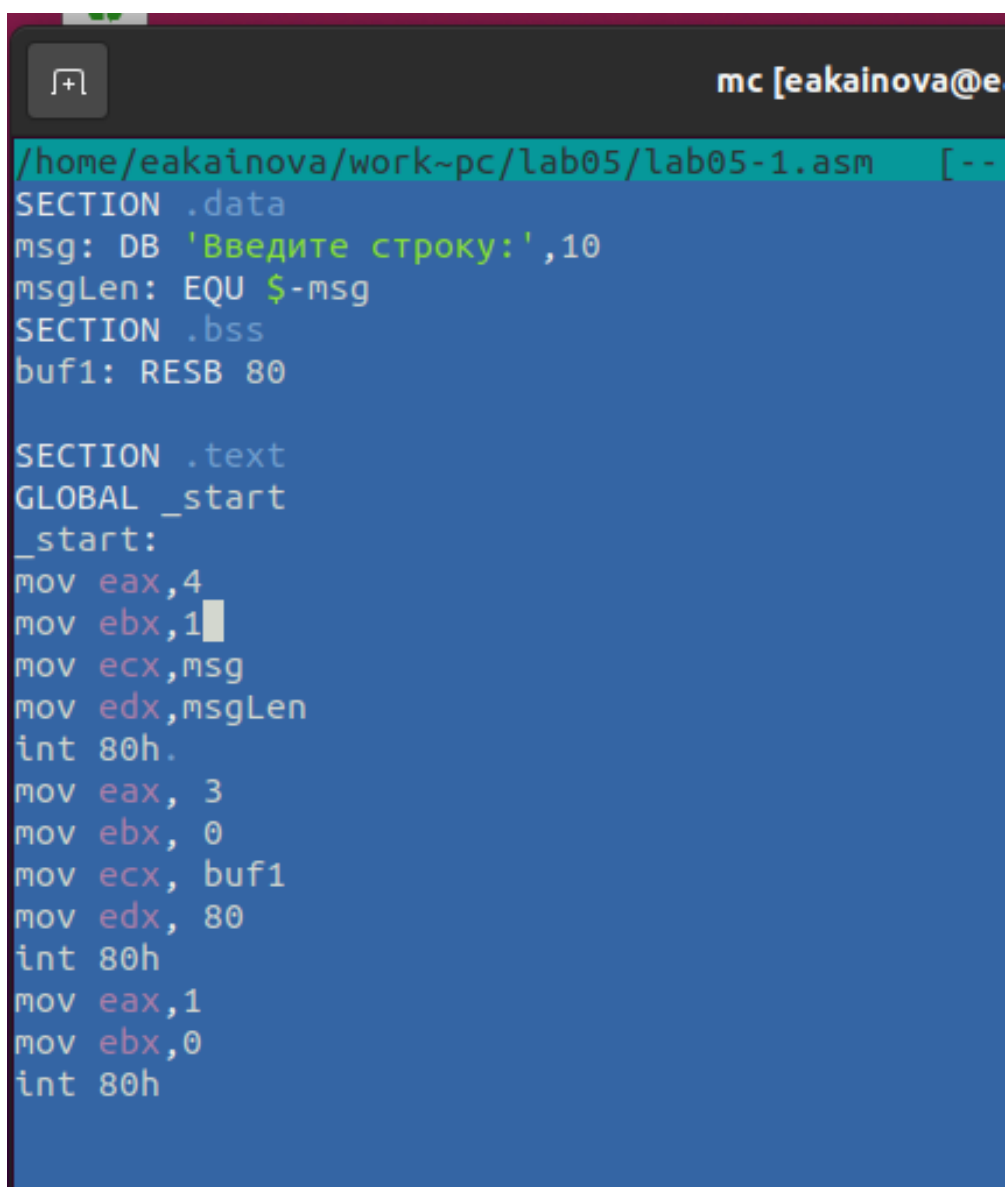
Открываю файл для редактирования с помощью F4, выбираю редактор mcedit, и пишу программу в соответствии с заданием (см. рис. 2.5).



A terminal window with a dark background. The title bar at the top shows a window icon and the text "mc [eakainova@eakainova]:~/work/arch-pc/lab05". The terminal content shows the user "eakainova@eakainova" at the prompt "~\$". They enter "mc". The prompt changes to "~/work/arch-pc/lab05\$". They enter "touch lab05-1.asm". Then, a menu appears: "Select an editor. To change later, run 'select-editor'." followed by a list: "1. /bin/nano <---- easiest", "2. /usr/bin/mcedit", "3. /usr/bin/vim.tiny", "4. /usr/bin/emacs", and "5. /bin/ed". At the bottom, it says "Choose 1-5 [1]: " with a cursor pointing to the space after "[1]".

```
mc [eakainova@eakainova]:~/work/arch-pc/lab05
eakainova@eakainova:~$ mc
eakainova@eakainova:~/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. 2.4: редактор mcedit

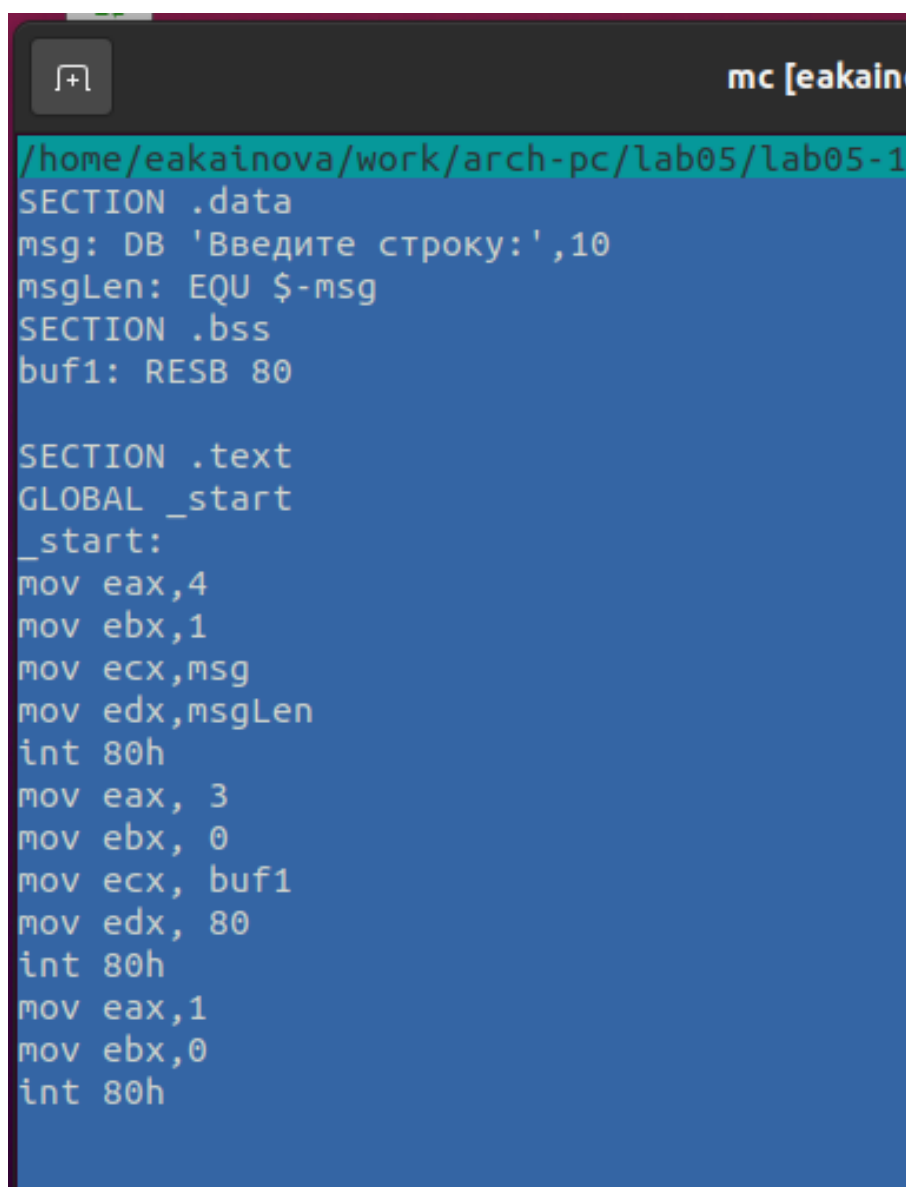


```
mc [eakainova@e
/home/eakainova/work~pc/lab05/lab05-1.asm  [ - -
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

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: Написание программы lab05-1.asm

Для проверки содержимого файла просматриваю его с помощью F3 и убеждаюсь в корректности написанного кода (см. рис. 2.6).



```
mc [eakainova]
/home/eakainova/work/arch-pc/lab05/lab05-1
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,1
mov ebx,0
int 80h
```

Рис. 2.6: Просмотр кода lab05-1.asm

Транслирую исходный код в объектный файл, затем выполняю компоновку для создания исполняемого файла программы (см. рис. 2.7).

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

Рис. 2.7: 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. 2.8). Для копирования использую F5, для перемещения — F6.

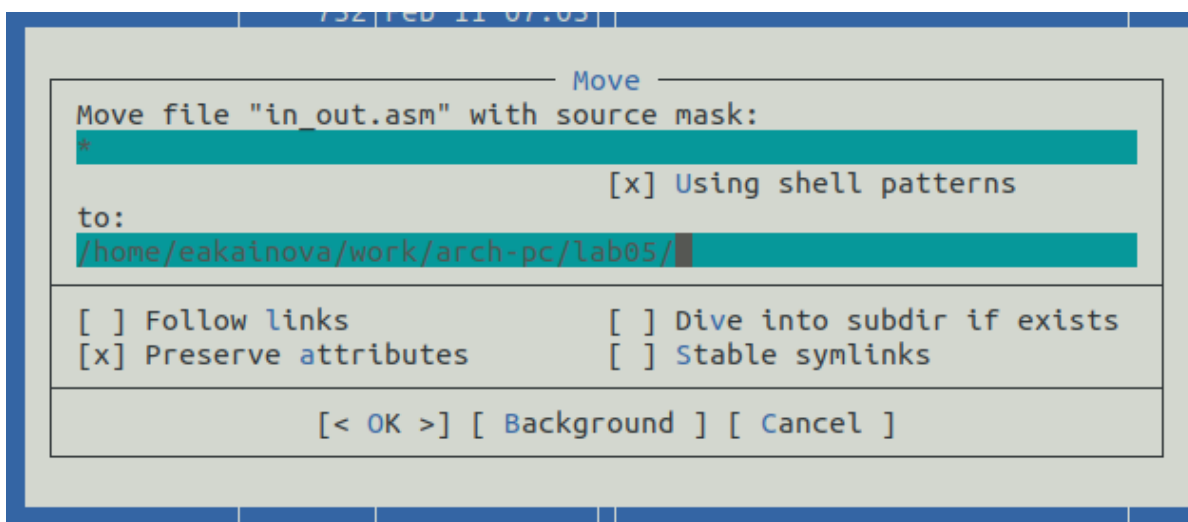


Рис. 2.8: Перемещение файла in\_out.asm

Копирую файл lab05-1.asm и создаю его копию под именем lab05-2.asm (см. рис. 2.9).

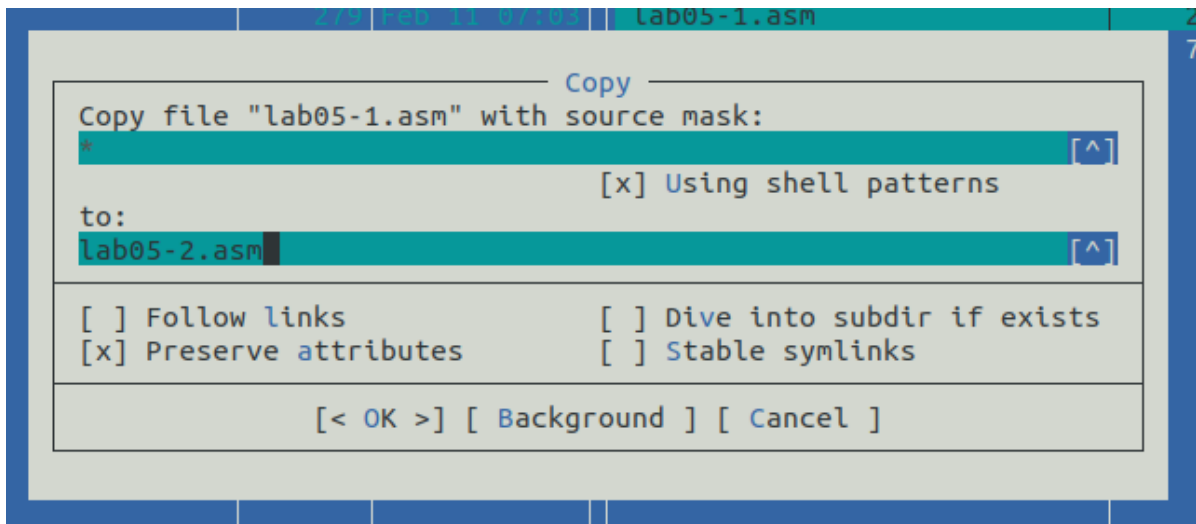
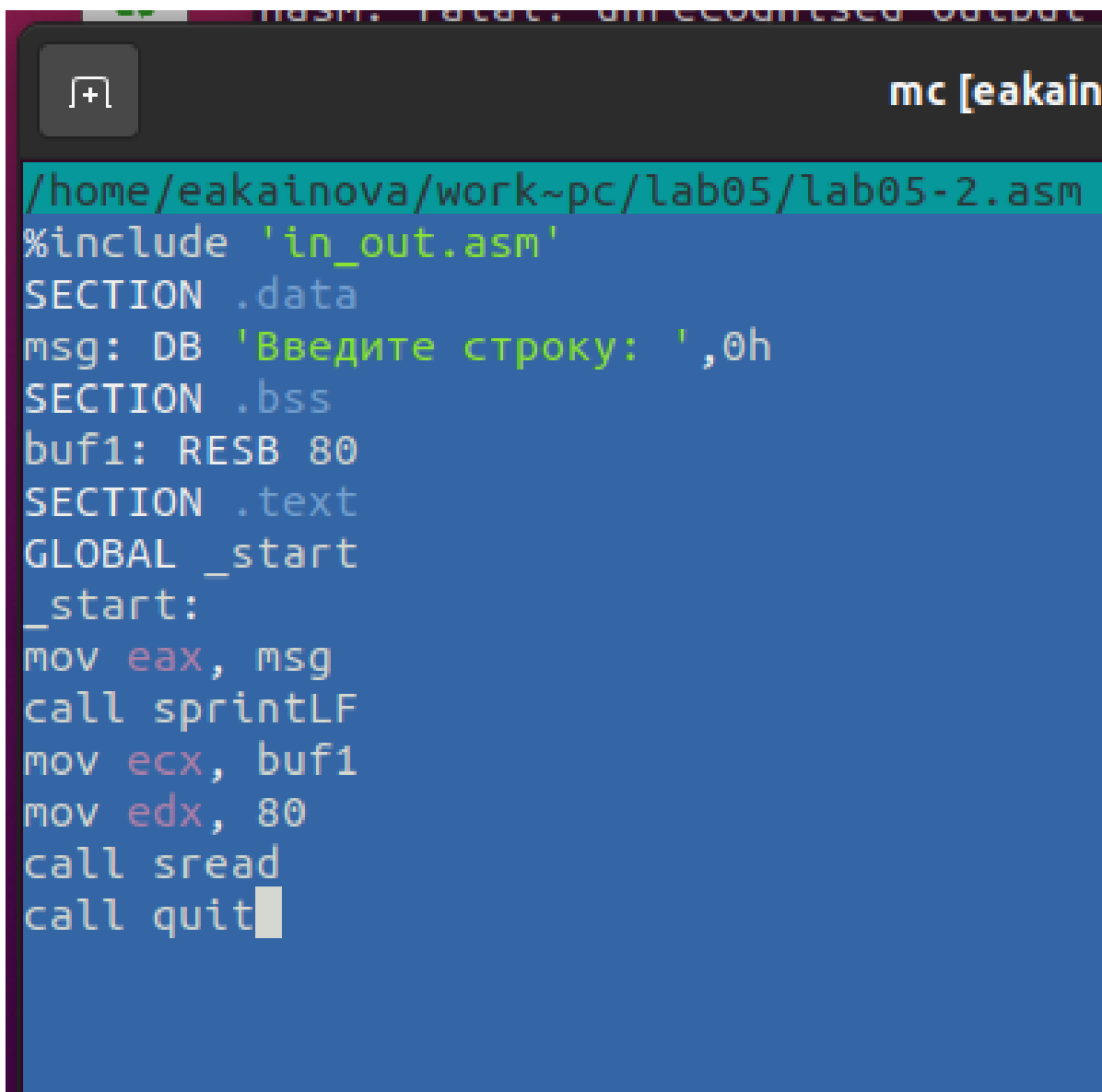


Рис. 2.9: Создание копии файла lab05-1.asm

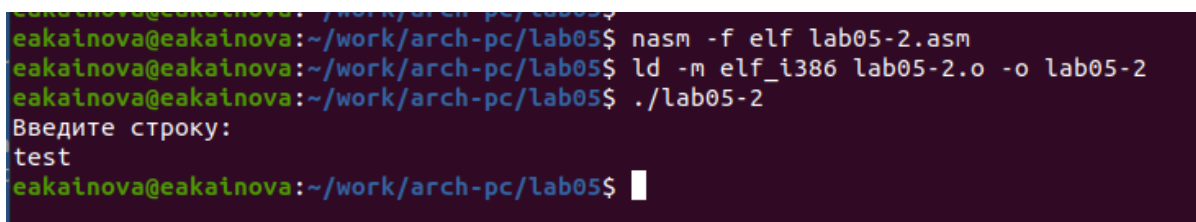
Редактирую код lab05-2.asm, добавляя подпрограммы из файла in\_out.asm (см. рис. 2.10).



```
mc [eakainova@eakainova: ~]$  
/home/eakainova/work~pc/lab05/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. 2.10: Добавление подпрограмм в lab05-2.asm

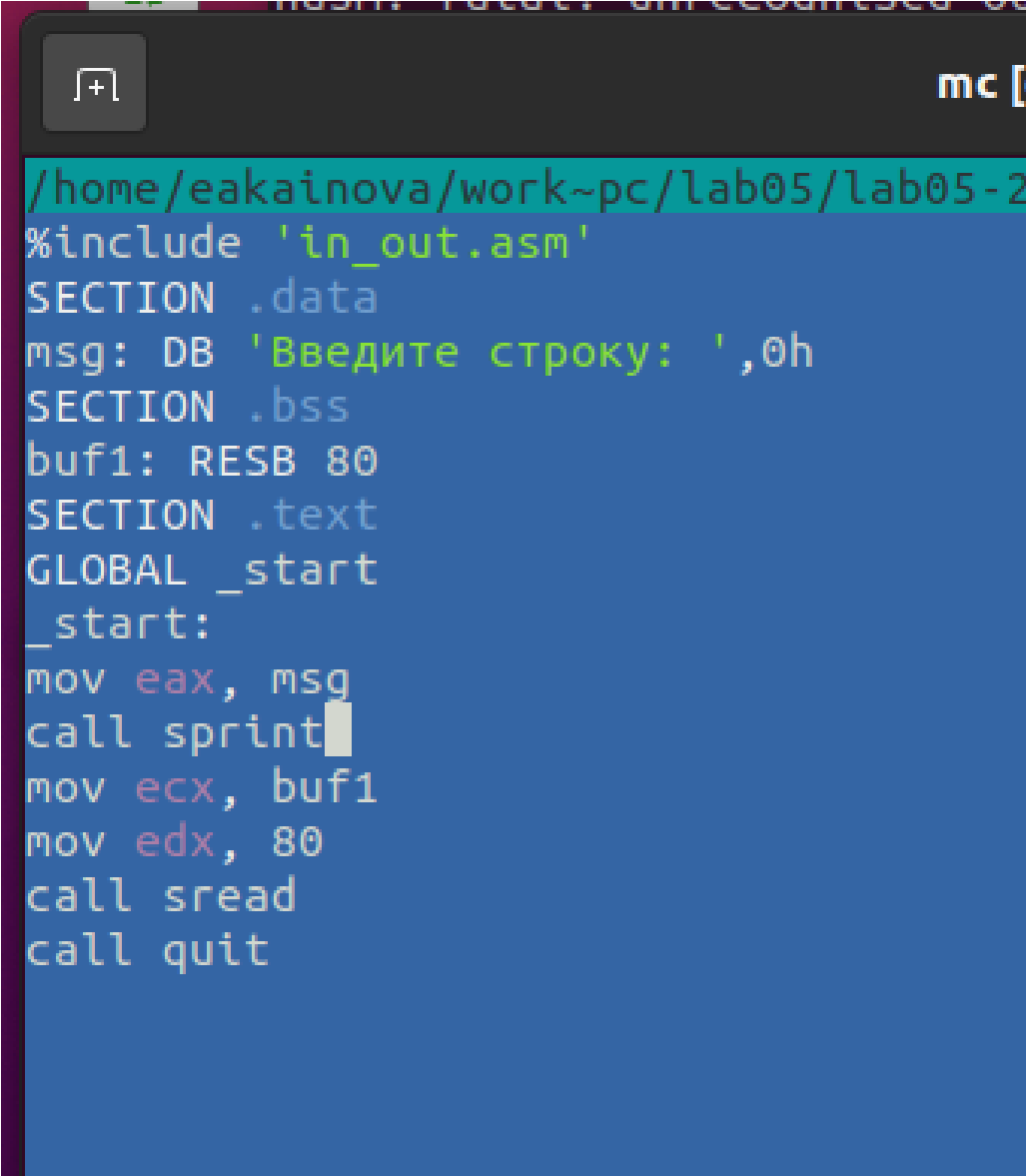
Компилирую программу и проверяю её выполнение (см. рис. 2.11).



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

Рис. 2.11: Запуск программы lab05-2.asm

В файле lab05-2.asm заменяю подпрограмму sprintLF на sprint, после чего повторно собираю программу (см. рис. 2.12 и 2.13).

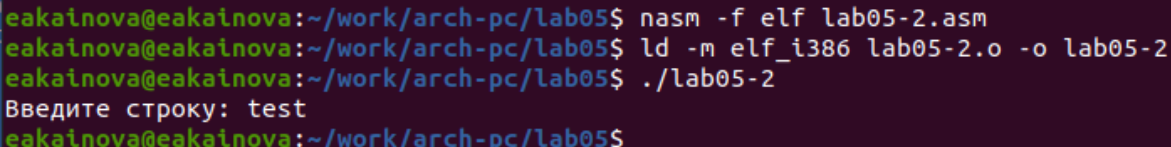


```

/home/eakainova/work~pc/lab05/lab05-2
#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 sprint
mov ecx, buf1
mov edx, 80
call sread
call quit

```

Рис. 2.12: Обновлённая версия lab05-2.asm



```

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

```

Рис. 2.13: 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. 2.14 и 2.15): - отображает запрос “Введите строку:”; - принимает строку с клавиатуры; - выводит введённую строку на экран.



```
CVDE nasm -n -f elf64
mc
/home/eakainova/work~pc/lab05/lab05-3.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,4
mov ebx,1
mov ecx,buf1
mov edx,80
int 80h
mov eax,1
mov ebx,0
int 80h
```

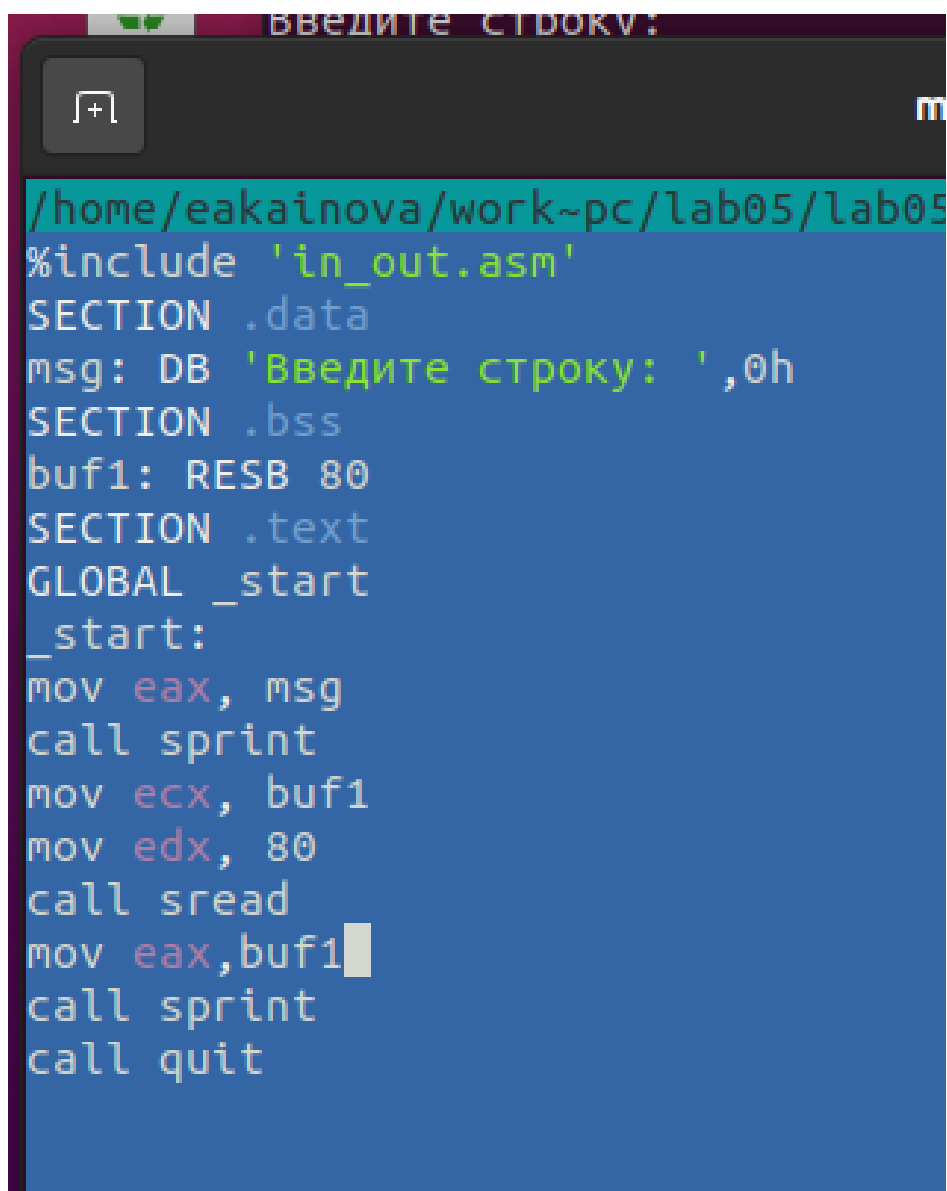
Рис. 2.14: Редактирование программы lab05-3.asm

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

Рис. 2.15: Тестовый запуск lab05-3.asm

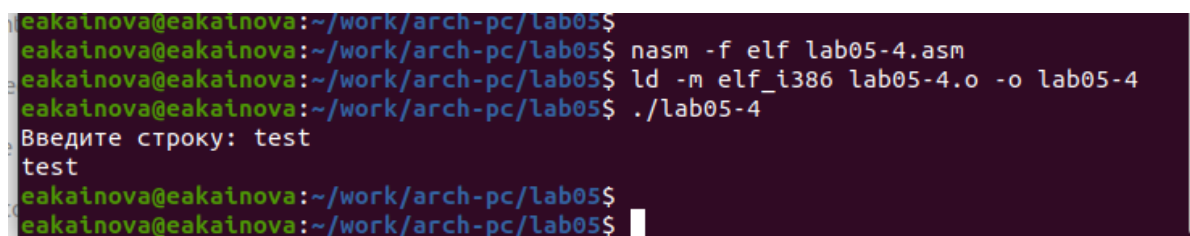
Аналогично, копирую lab05-2.asm и изменяю код, добавляя подпрограммы из

файла in\_out.asm (см. рис. 2.16 и 2.17).



```
ВВЕДИТЕ СТРОКУ:
/home/eakainova/work~pc/lab05/lab05
%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. 2.16: Обновлённая программа lab05-4.asm



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

Рис. 2.17: Тестовый запуск lab05-4.asm

## 3 Выводы

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