

Отчёт по лабораторной работе 5

дисциплина: Архитектура компьютера

Магомедов Султан Гасанович

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	13
2.3	Задание для самостоятельной работы	17
3	Выводы	19

Список иллюстраций

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

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).

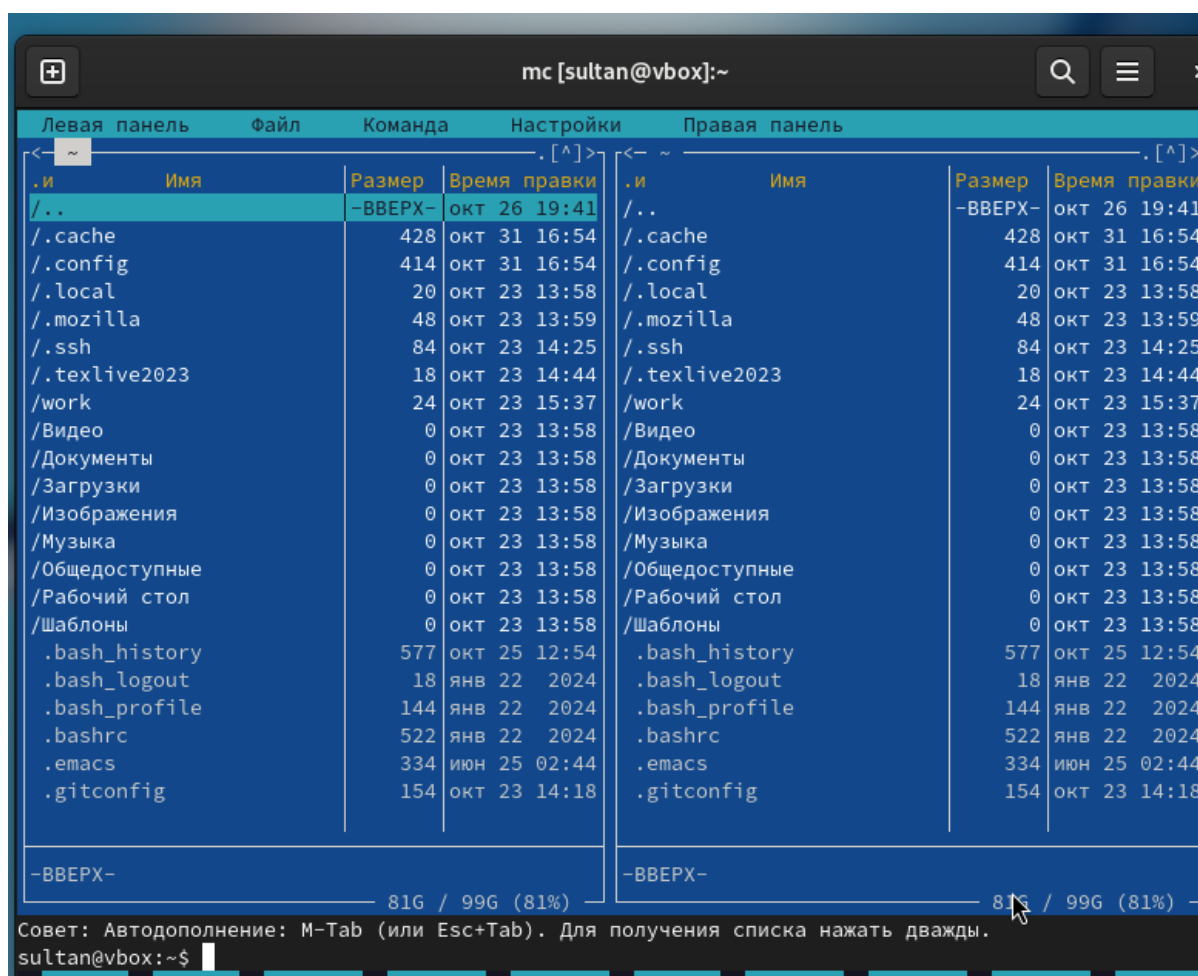


Рис. 2.1: Запуск Midnight Commander

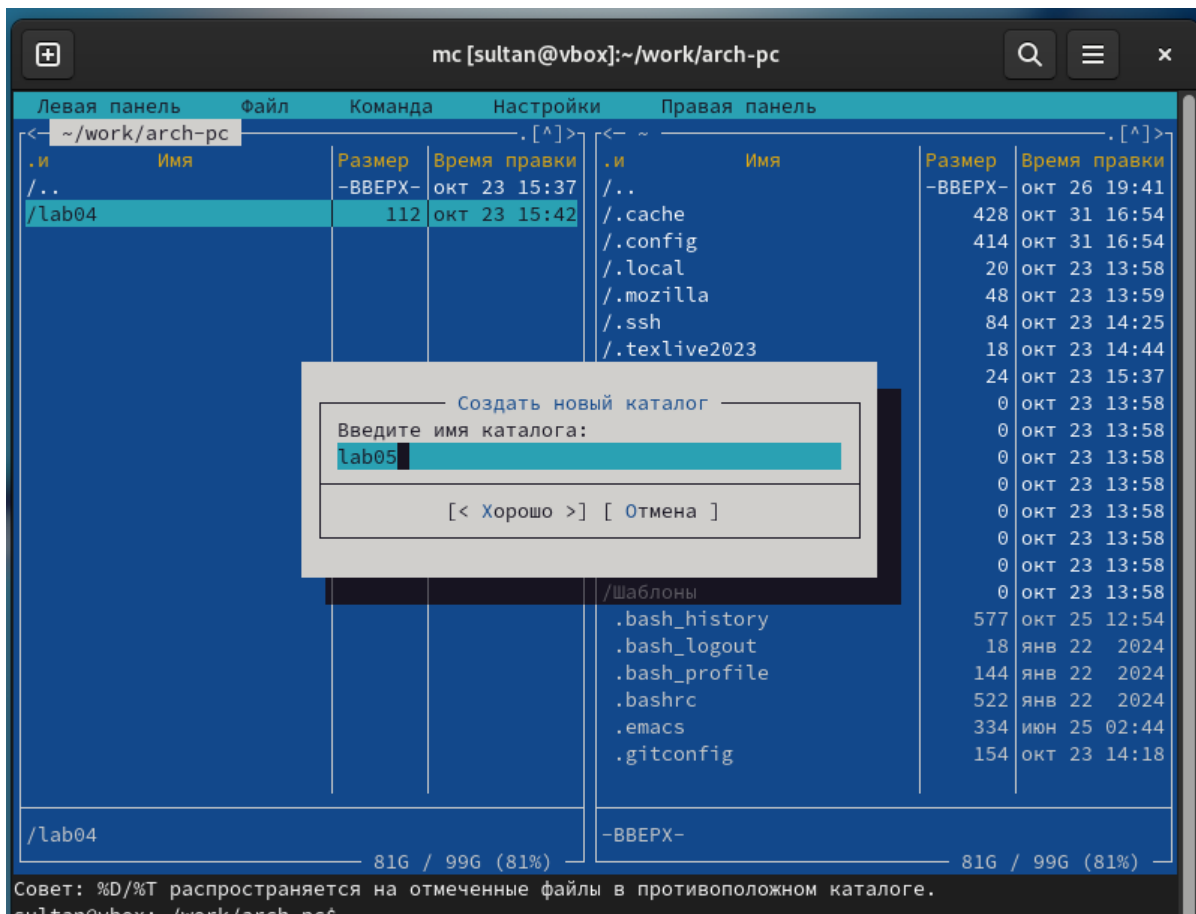


Рис. 2.2: Создание нового каталога

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

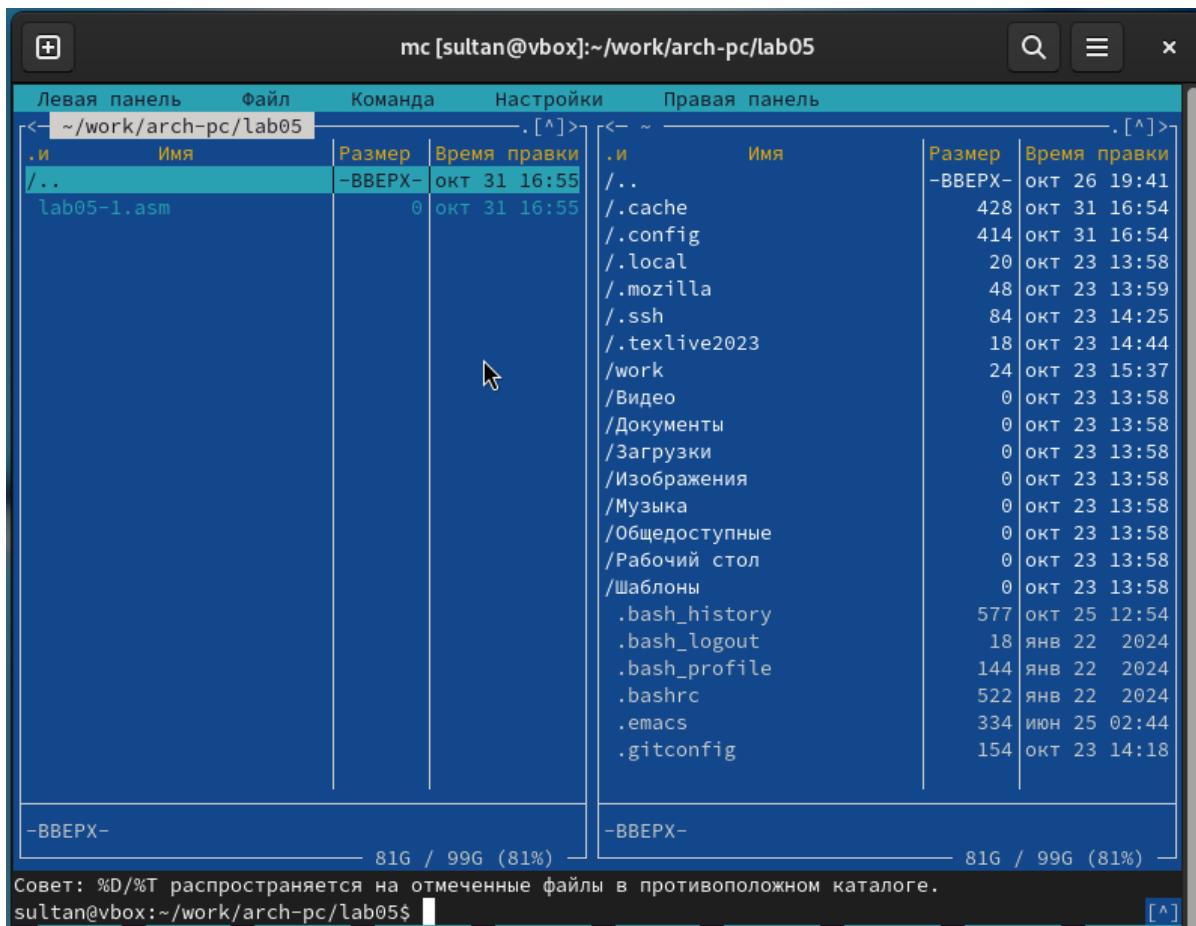


Рис. 2.3: Создание файла lab05-1.asm

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

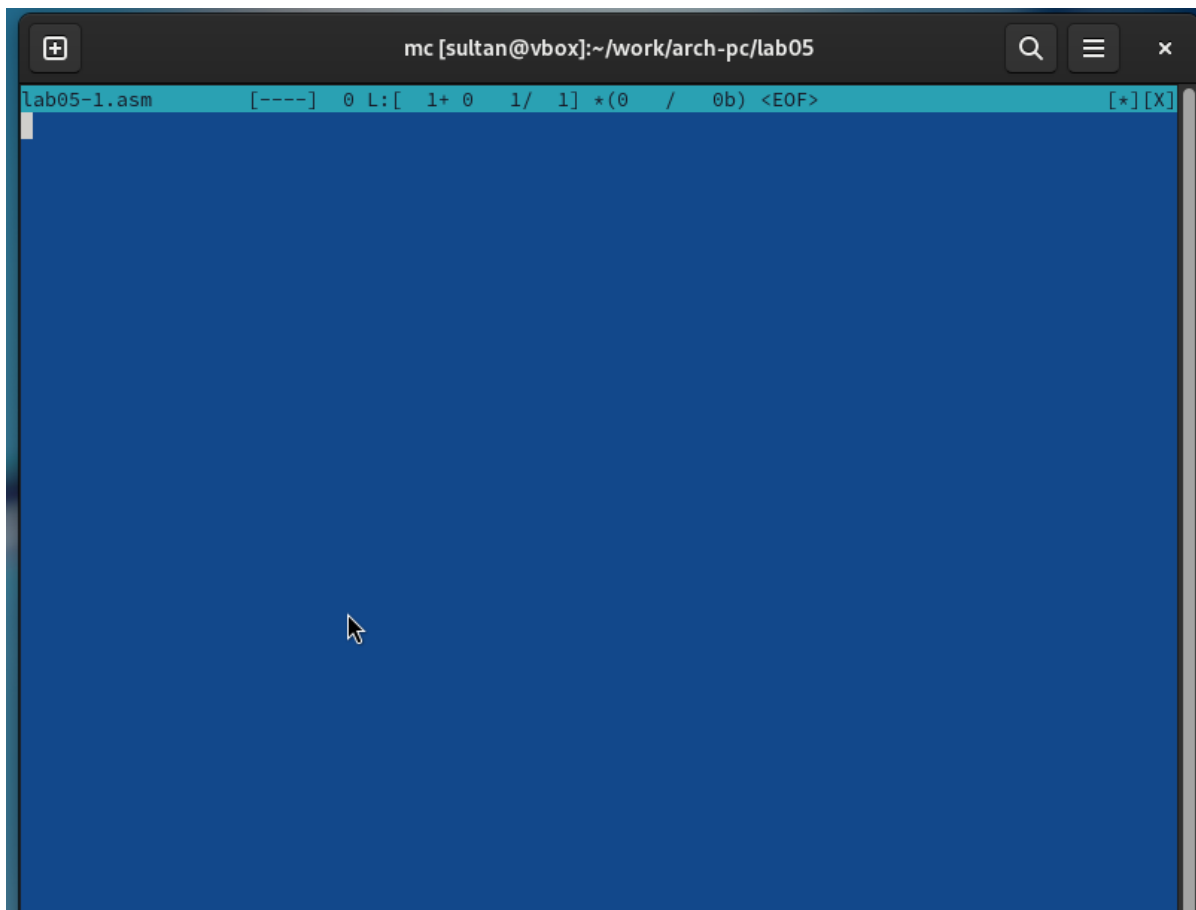
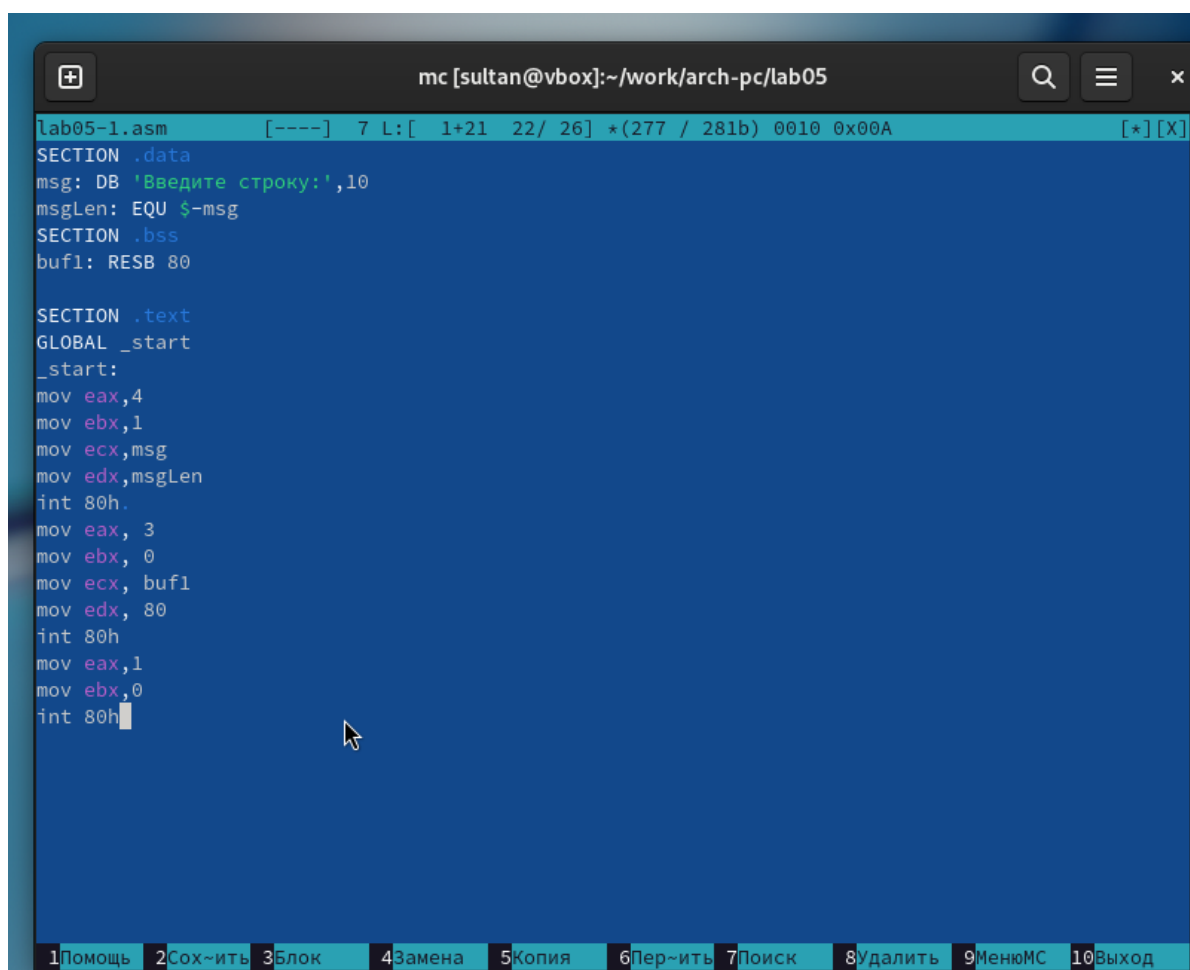


Рис. 2.4: Выбор редактора mcedit

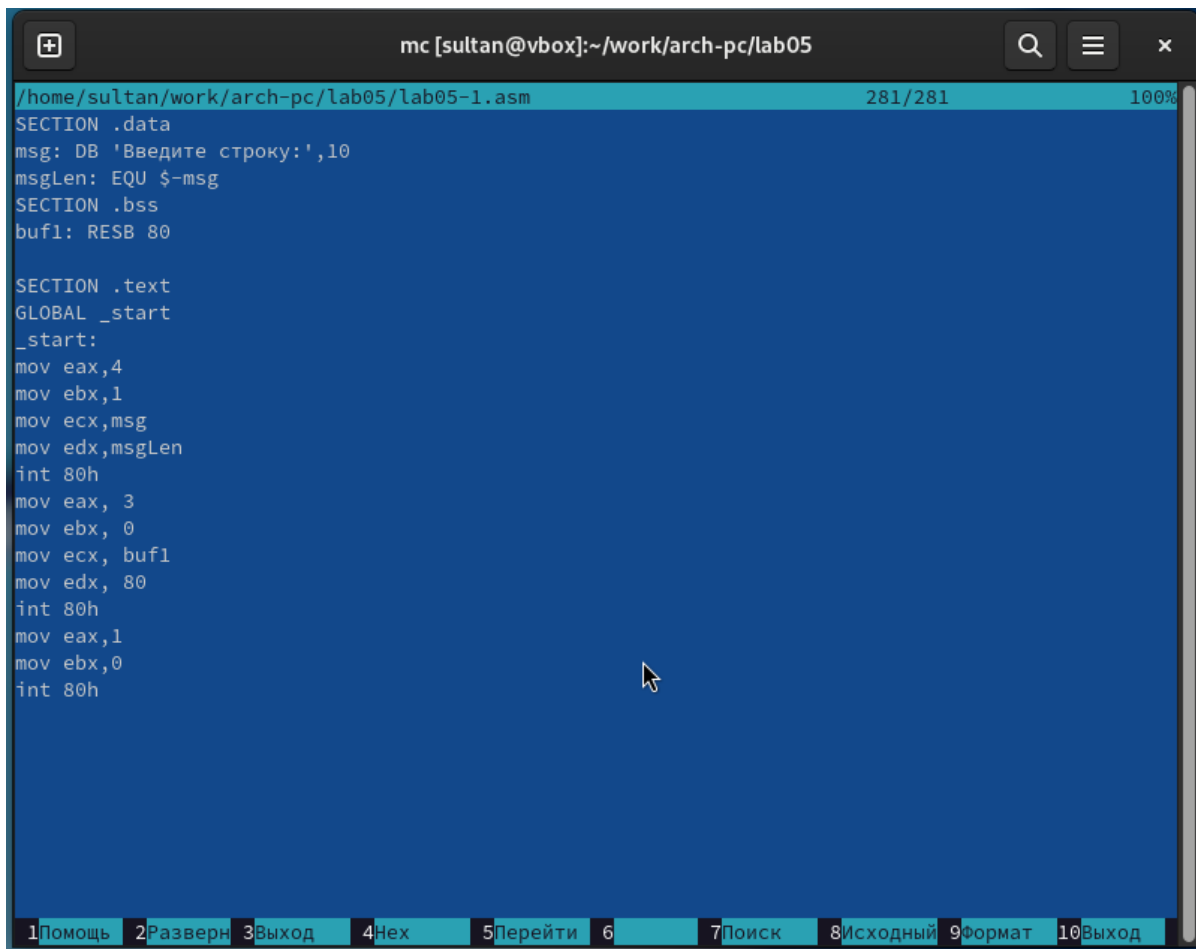


```
lab05-1.asm      [----]  7 L:[  1+21  22/ 26] *(277 / 281b) 0010 0x00A [*] [X]  
SECTION .data  
msg: DB 'Введите строку:',10  
msglen: EQU $-msg  
SECTION .bss  
buf1: RESB 80  
  
SECTION .text  
GLOBAL _start  
_start:  
mov eax,4  
mov ebx,1  
mov ecx,msg  
mov edx,msglen  
int 80h  
mov eax, 3  
mov ebx, 0  
mov ecx, buf1  
mov edx, 80  
int 80h  
mov eax,1  
mov ebx,0  
int 80h
```

1Помощь 2Сох-ить 3Блок 4Замена 5Копия 6Пер-ить 7Поиск 8Удалить 9МенюМС 10Выход

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).



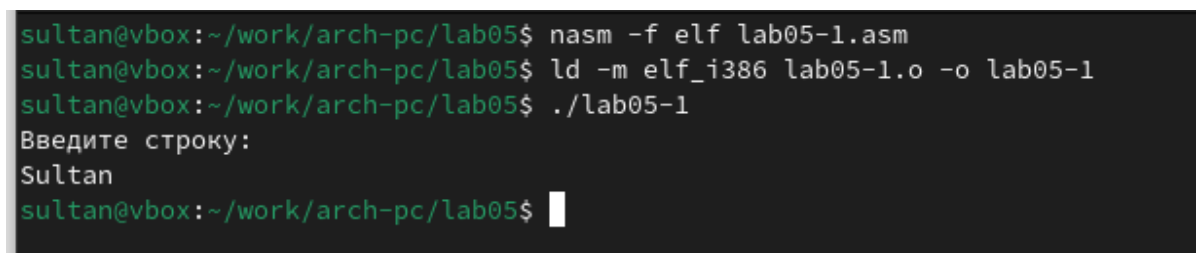
```
mc [sultan@vbox]:~/work/arch-pc/lab05
/home/sultan/work/arch-pc/lab05/lab05-1.asm 281/281 100%
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

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
```

1Помощь 2Развернуть 3Выход 4Hex 5Перейти 6 7Поиск 8Исходный 9Формат 10Выход

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).



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

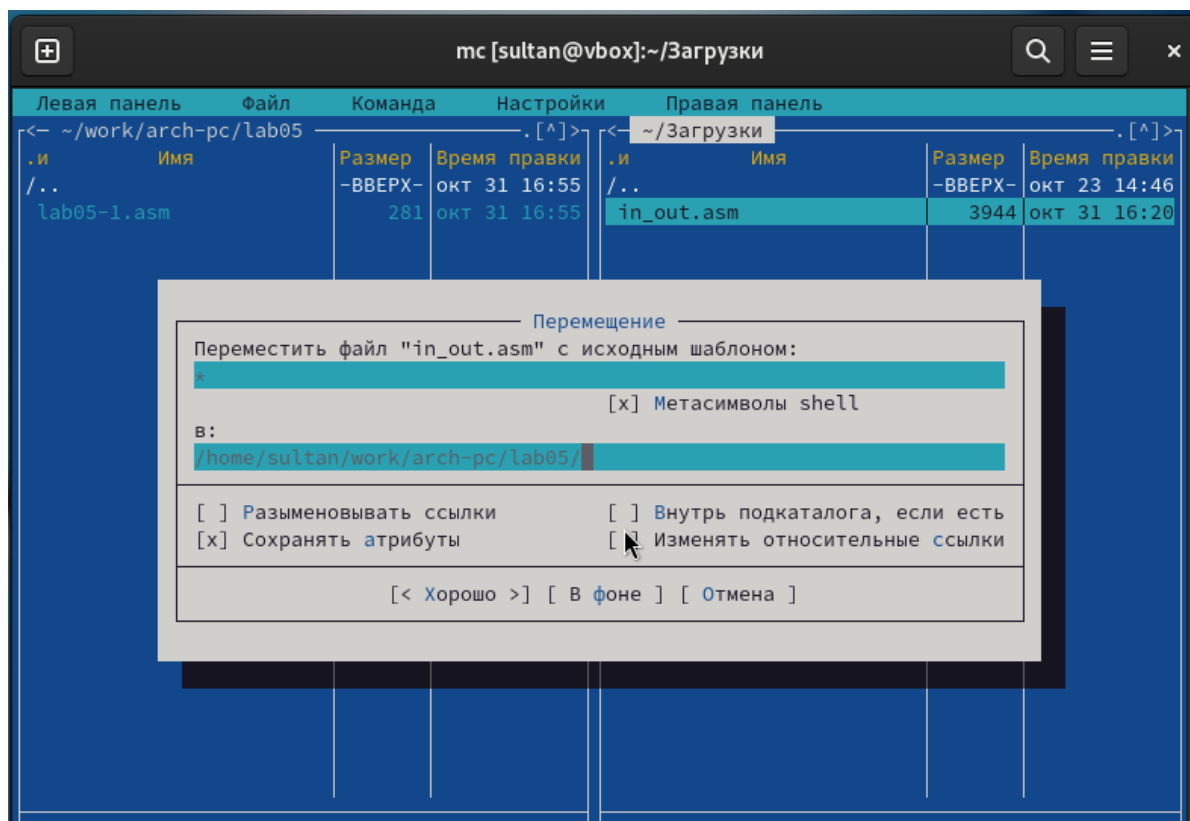


Рис. 2.8: Перемещение файла `in_out.asm`

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

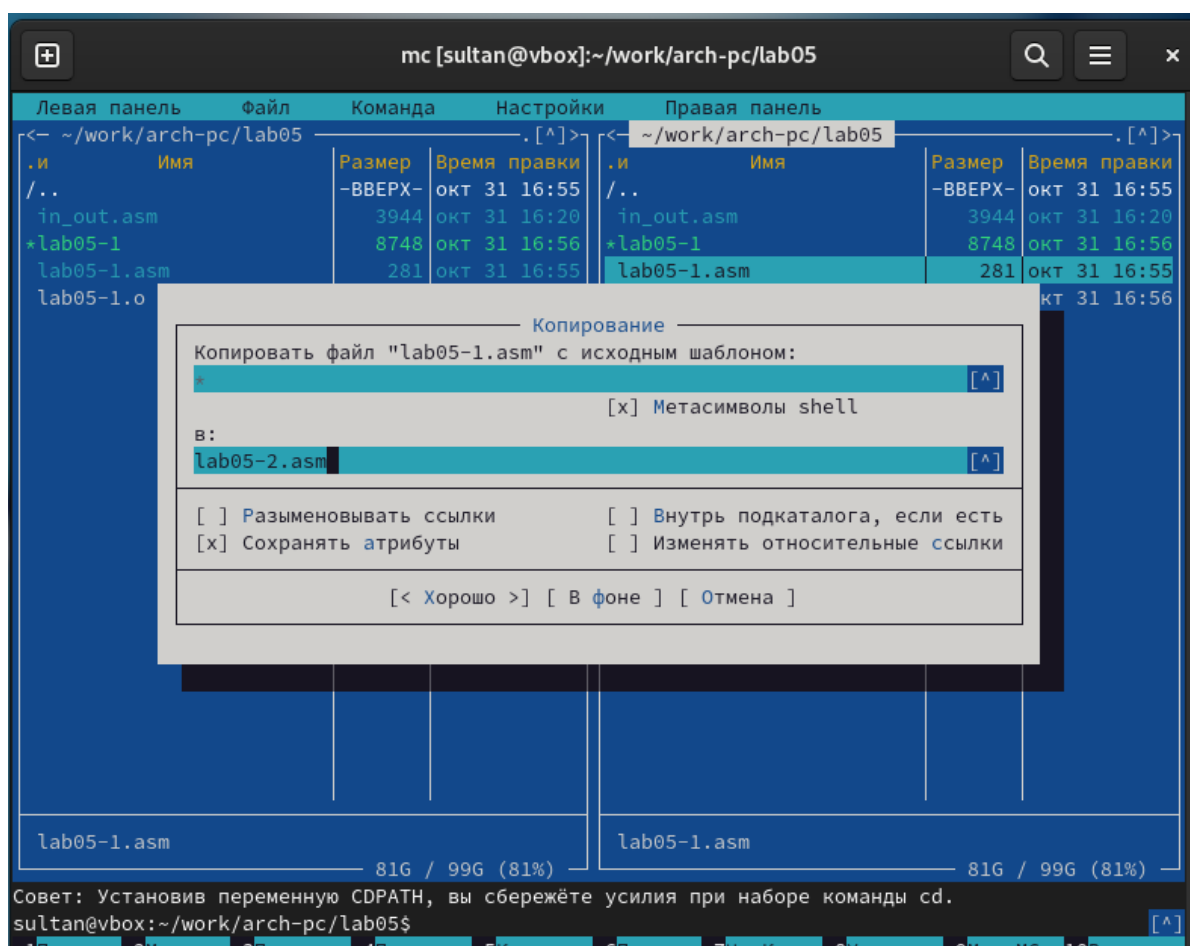
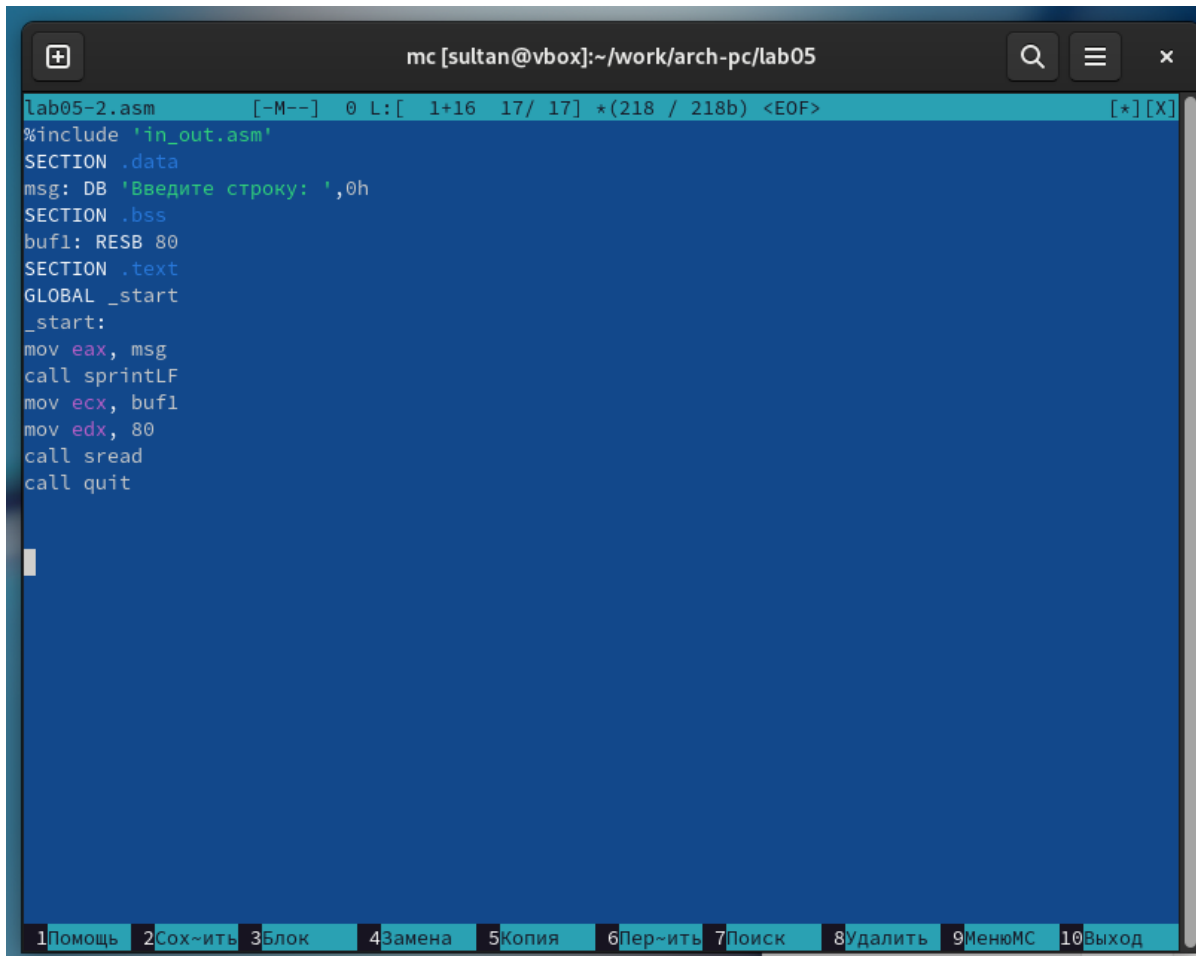


Рис. 2.9: Создание копии файла lab05-1.asm

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

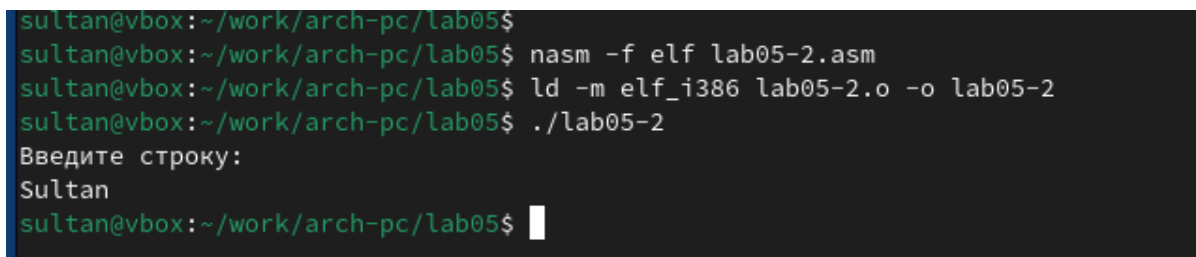


```
lab05-2.asm [-M--] 0 L: [ 1+16 17/ 17] *(218 / 218b) <EOF> [*] [X]
#include 'in_out.asm'
SECTION .data
msg: DB 'Введите строку: ',0h
SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:
mov eax, msg
call sprintLF
mov ecx, buf1
mov edx, 80
call sread
call quit
```

1Помощь 2Сохранить 3Блок 4Замена 5Копия 6Перенести 7Поиск 8Удалить 9МенюМС 10Выход

Рис. 2.10: Добавление подпрограмм в lab05-2.asm

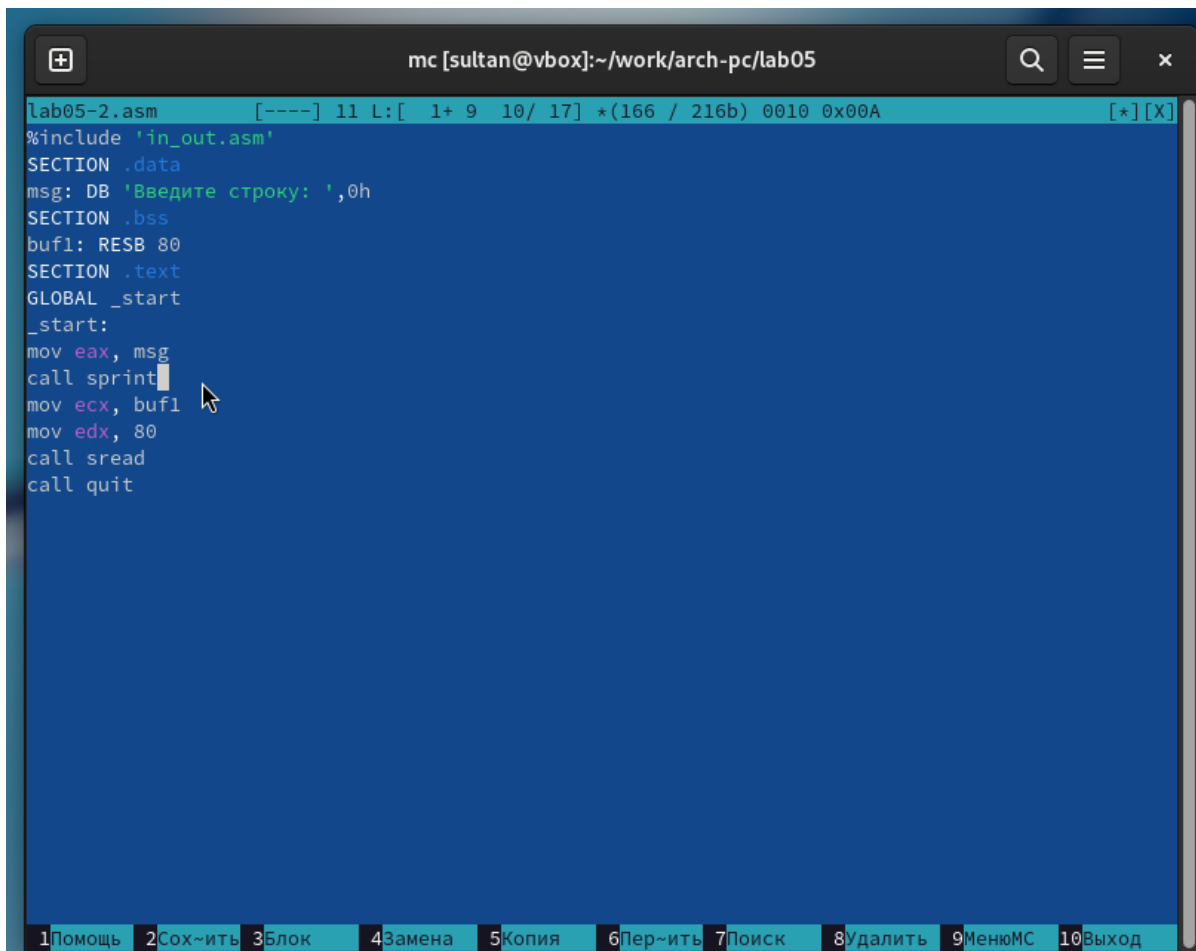
Компилирую программу и проверяю её выполнение (см. рис. 2.11).



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

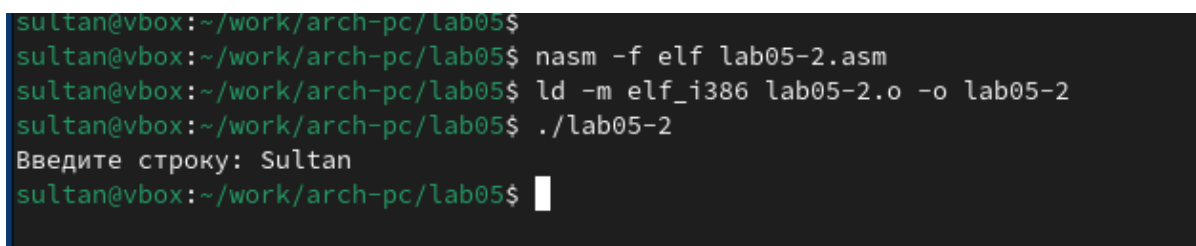
Рис. 2.11: Запуск программы lab05-2.asm

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



```
lab05-2.asm [---] 11 L: [ 1+ 9 10/ 17] *(166 / 216b) 0010 0x00A [*][X]
#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



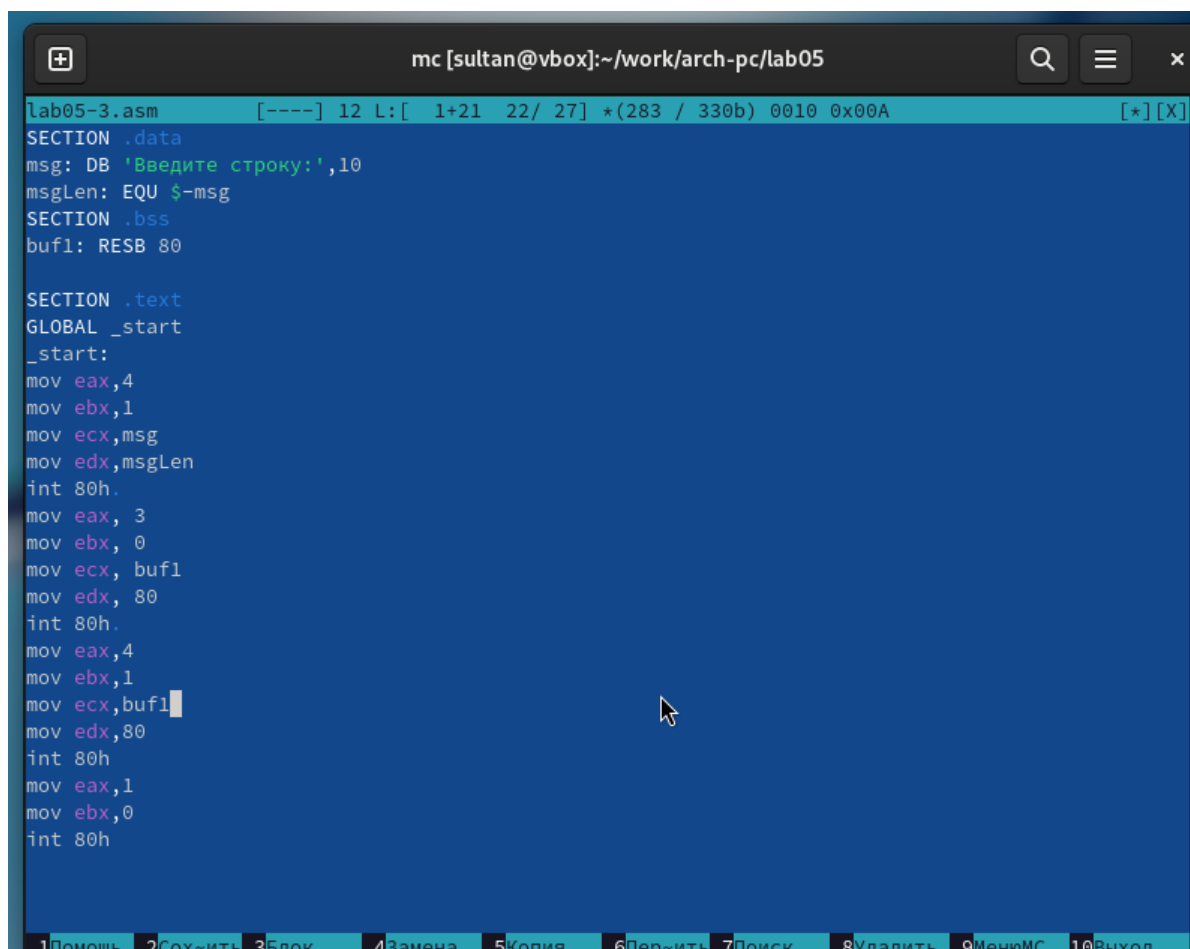
```
sultan@vbox:~/work/arch-pc/lab05$
sultan@vbox:~/work/arch-pc/lab05$ nasm -f elf lab05-2.asm
sultan@vbox:~/work/arch-pc/lab05$ ld -m elf_i386 lab05-2.o -o lab05-2
sultan@vbox:~/work/arch-pc/lab05$ ./lab05-2
Введите строку: Sultan
sultan@vbox:~/work/arch-pc/lab05$
```

Рис. 2.13: Запуск обновлённой lab05-2.asm

Теперь программа выводит строку без переноса на новую строку.

2.3 Задание для самостоятельной работы

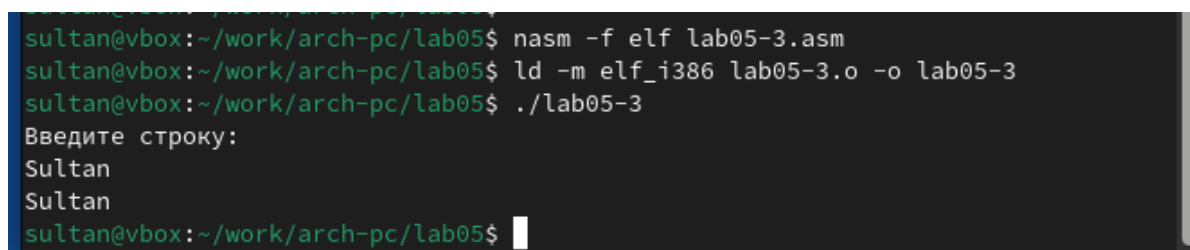
Создаю копию программы lab05-1.asm и изменяю код для выполнения следующего алгоритма (см. рис. 2.14 и 2.15): - отображает запрос “Введите строку:”; - принимает строку с клавиатуры; - выводит введенную строку на экран.



```
lab05-3.asm [----] 12 L: [ 1+21 22/ 27] *(283 / 330b) 0010 0x00A [*] [X]
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
```

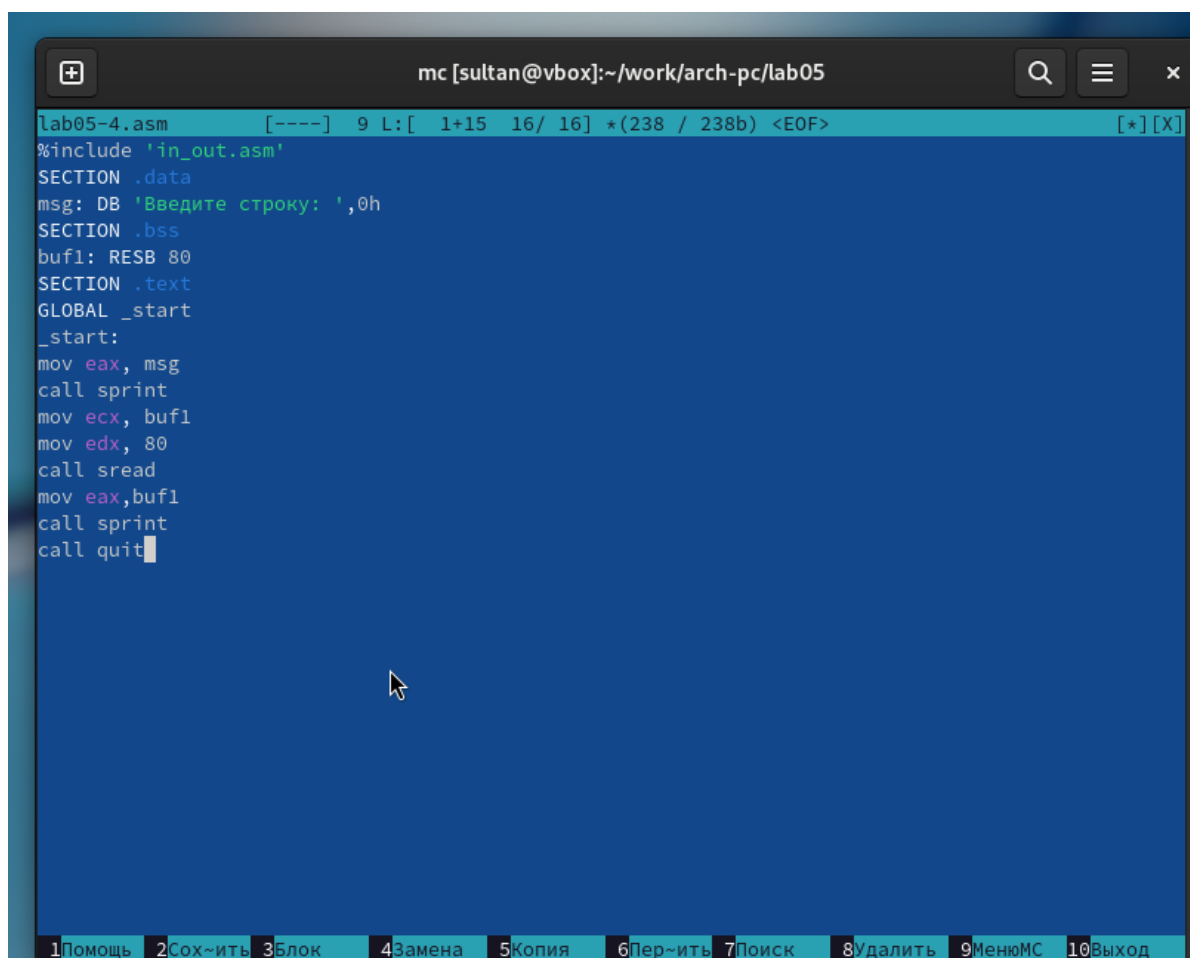
Рис. 2.14: Редактирование программы lab05-3.asm



```
sultan@vbox:~/work/arch-pc/lab05$ nasm -f elf lab05-3.asm
sultan@vbox:~/work/arch-pc/lab05$ ld -m elf_i386 lab05-3.o -o lab05-3
sultan@vbox:~/work/arch-pc/lab05$ ./lab05-3
Введите строку:
Sultan
Sultan
sultan@vbox:~/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).



```
lab05-4.asm [----] 9 L: [ 1+15 16/ 16] *(238 / 238b) <EOF> [*] [X]
#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



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