

Отчёт по лабораторной работе 5

дисциплина: Архитектура компьютера

Саидмагомедов Джамаль

Содержание

1	Цель работы	5
2	Выполнение лабораторной работы	6
3	Выводы	15

Список иллюстраций

2.1	Создание каталога	6
2.2	Программа lab05-1.asm	7
2.3	Просмотр файла lab05-1.asm	8
2.4	Запуск программы lab05-1.asm	9
2.5	Копирование файла	9
2.6	Программа lab05-2.asm	10
2.7	Запуск программы lab05-2.asm	10
2.8	Программа lab05-2.asm	11
2.9	Запуск программы lab05-2.asm	11
2.10	Программа lab05-3.asm	12
2.11	Запуск программы lab05-3.asm	13
2.12	Программа lab05-4.asm	14
2.13	Запуск программы lab05-4.asm	14

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

1. Открыл Midnight Commander. Перешел в каталог ~/work/arch-pc. Создал каталог lab05.

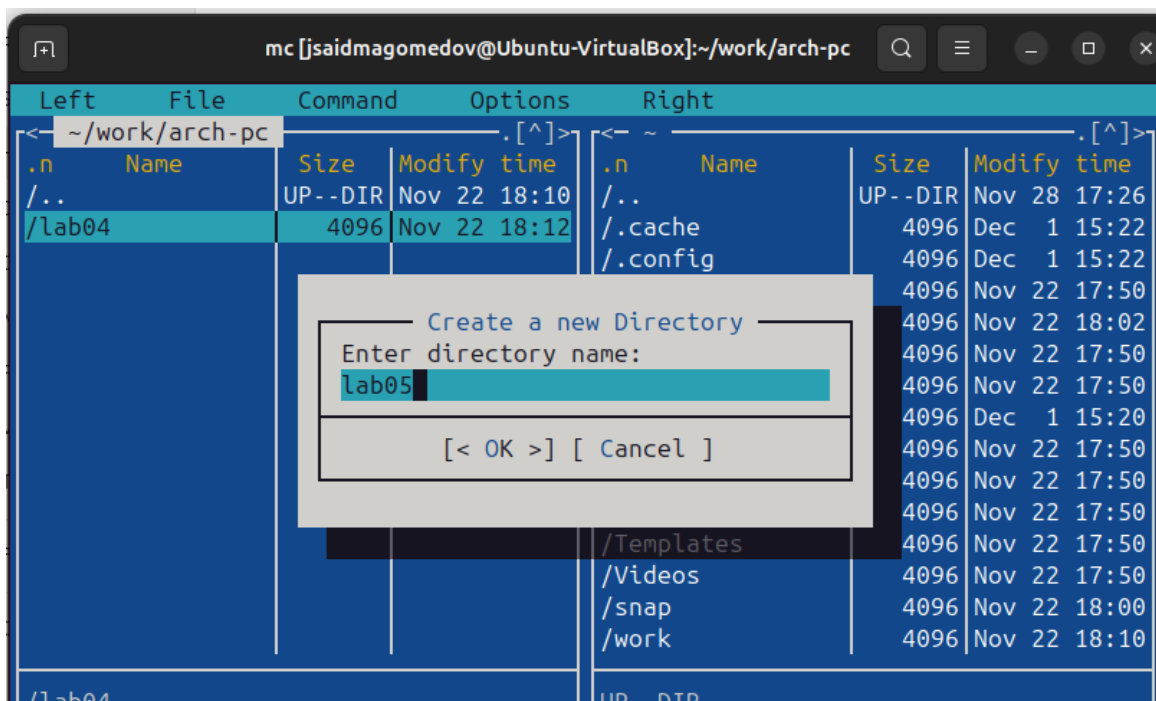
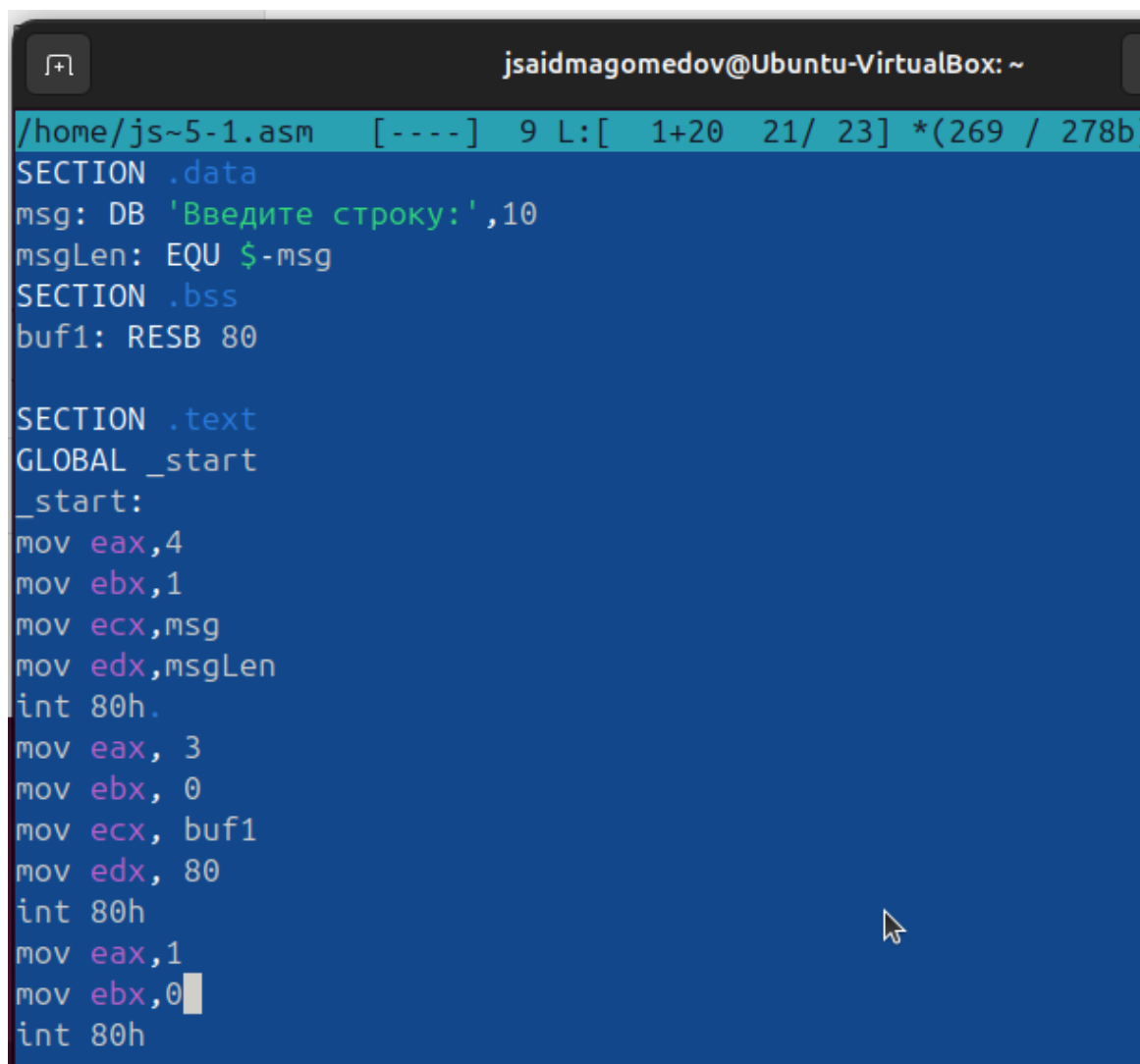


Рисунок 2.1: Создание каталога

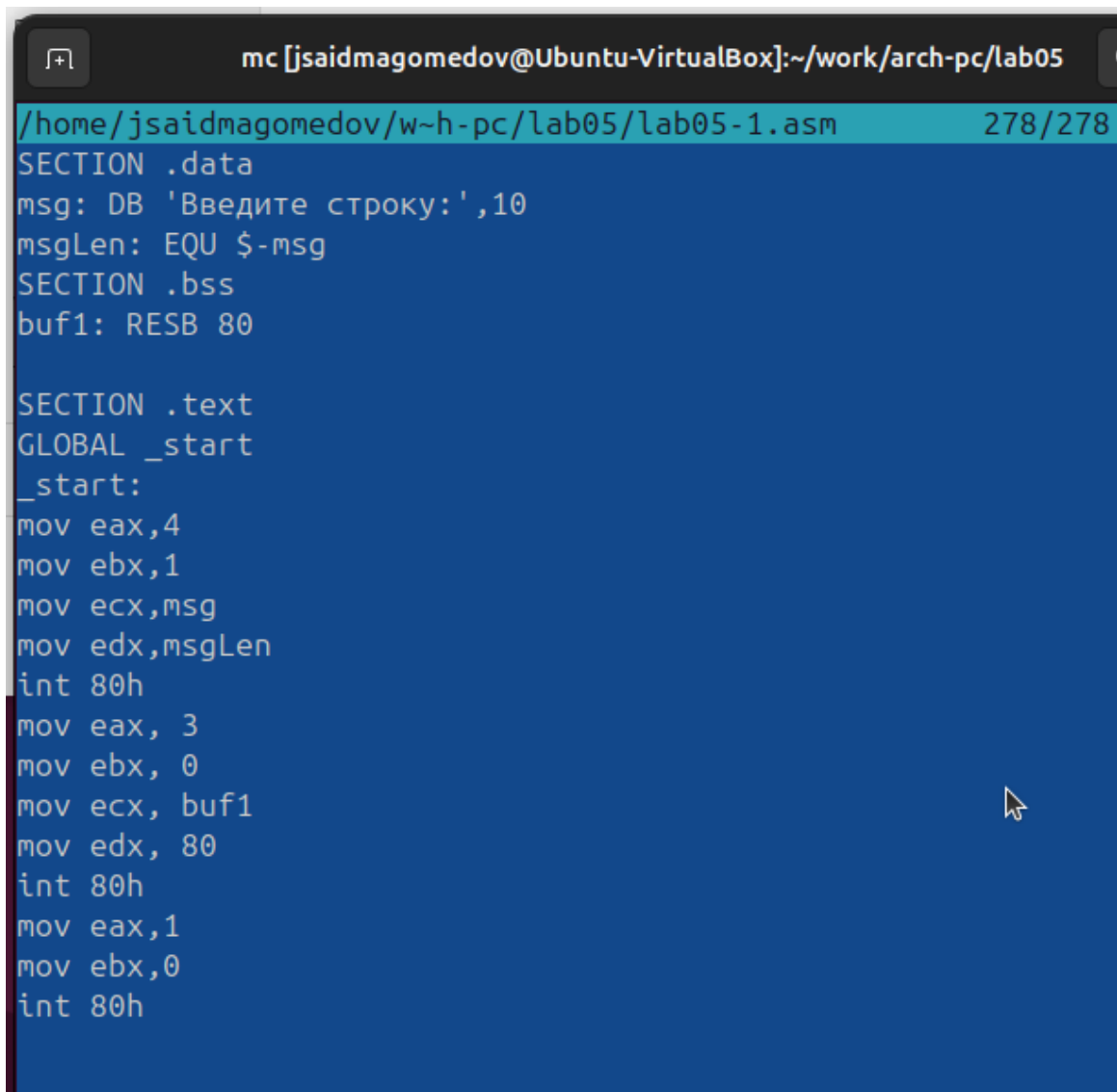
2. Создал файл lab05-1.asm. Открыл файл на редактирование и написал код.



```
jsaidmagomedov@Ubuntu-VirtualBox: ~  
/home/js~5-1.asm [----] 9 L:[ 1+20 21/ 23] *(269 / 278b  
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 2.2: Программа lab05-1.asm

3. Открыл файл на просмотр и проверил набранный код.

A screenshot of a code editor window. The title bar shows the path `mc [jsaidmagomedov@Ubuntu-VirtualBox]:~/work/arch-pc/lab05`. The editor displays the contents of `lab05-1.asm`, with the file name and line count `278/278` in the top right. The code is written in assembly language and is organized into sections: `.data`, `.bss`, and `.text`. The `.data` section defines a message string and its length. The `.bss` section reserves space for a buffer. The `.text` section contains the main logic, starting with a global `_start` function. The code uses `mov` instructions to set up registers and `int 80h` for system calls. A mouse cursor is visible on the right side of the code area.

```
mc [jsaidmagomedov@Ubuntu-VirtualBox]:~/work/arch-pc/lab05
/home/jsaidmagomedov/w~h-pc/lab05/lab05-1.asm 278/278
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
```

Рисунок 2.3: Просмотр файла lab05-1.asm

4. Получил исполняемый файл и проверил как он работает.


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

Рисунок 2.4: Запуск программы lab05-1.asm

5. Скачал файл in_out.asm. Добавил файл in_out.asm в рабочий каталог. Скопировал lab05-1.asm в lab05-2.asm.

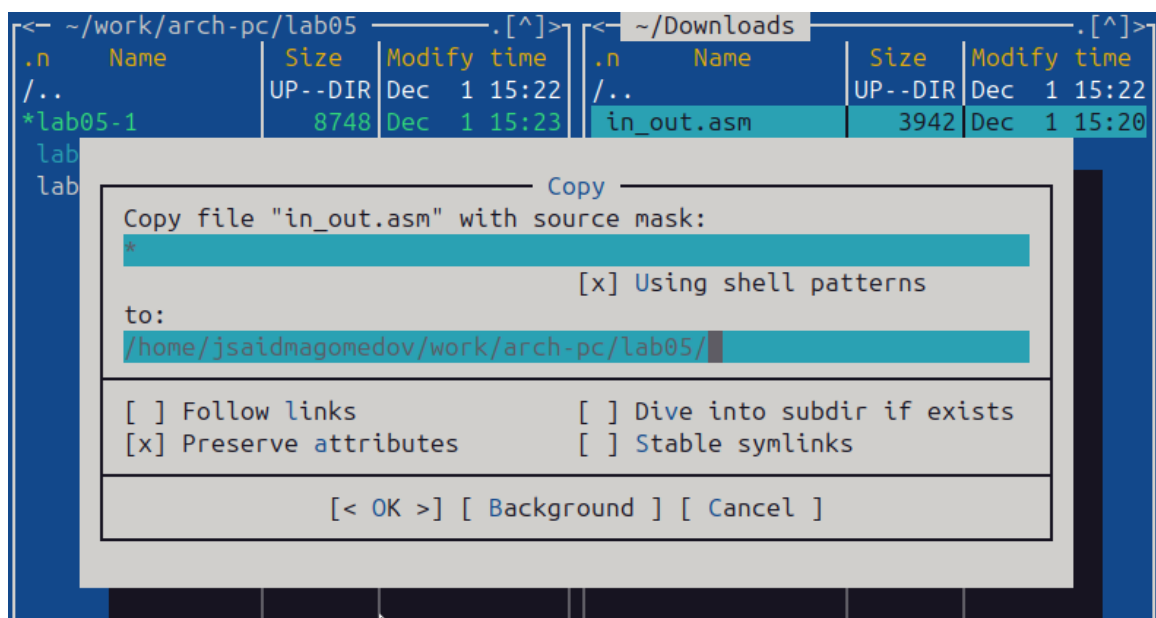


Рисунок 2.5: Копирование файла

6. Написал код программы lab05-2.asm. Скомпилировал программу и протестировал запуск.

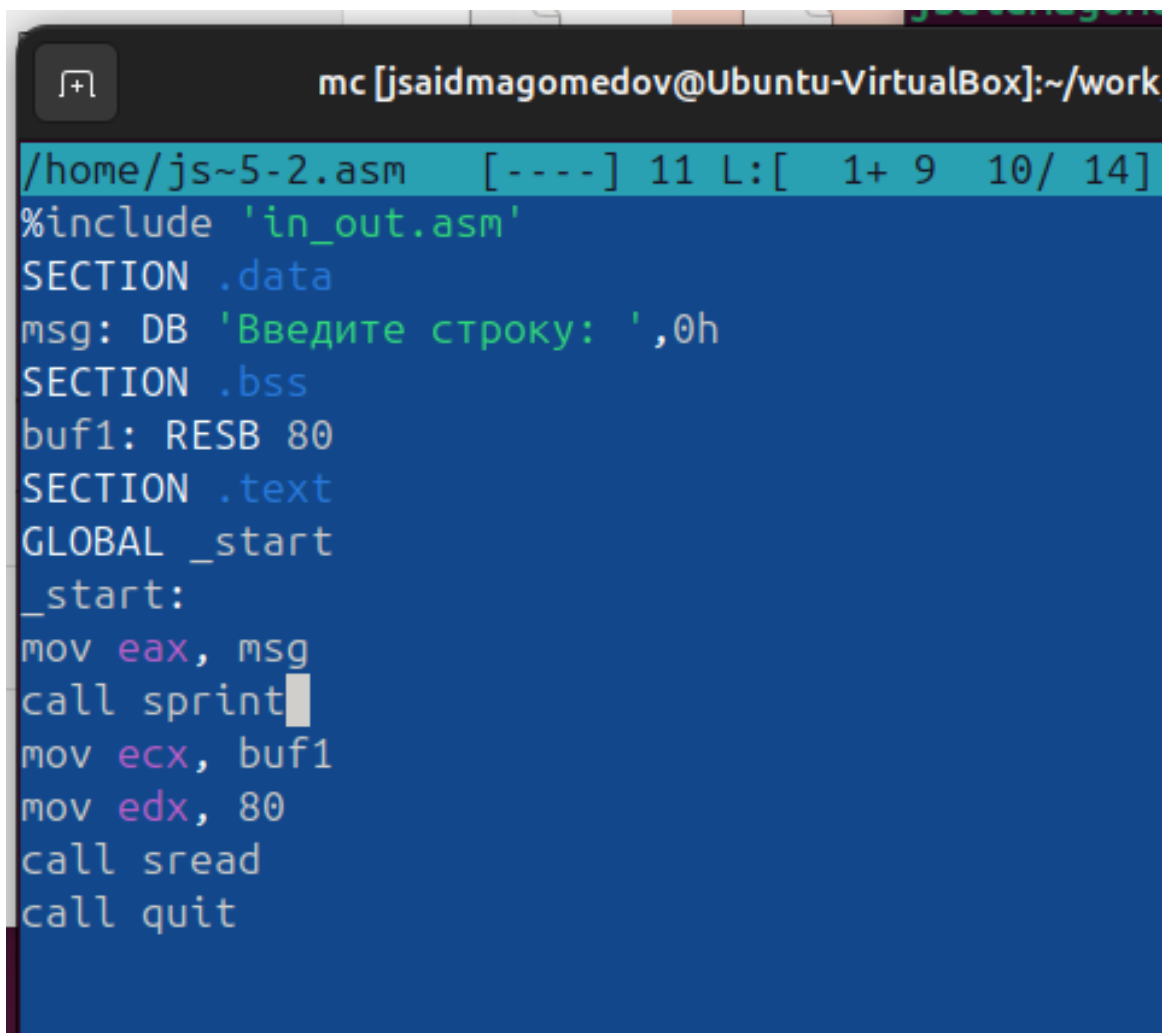
```
mc [jsaidmagomedov@Ubuntu-VirtualBox]:~/work/arch-pc/lab05
/home/js~5-2.asm  [ - - - - ] 10 L: [ 1+12 13/ 14] *(205 / 21
%include 'in_out.asm'
SECTION .data
msg: DB 'Введите строку: ',0h
SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:
mov eax, msg
call sprintf
mov ecx, buf1
mov edx, 80
call sread
call quit
```

Рисунок 2.6: Программа lab05-2.asm

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

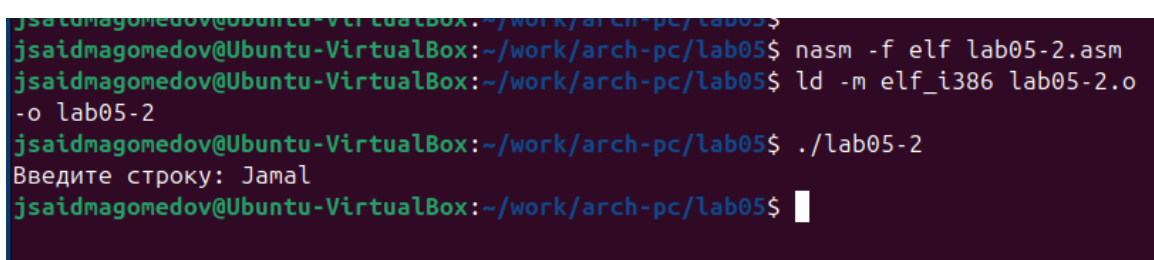
Рисунок 2.7: Запуск программы lab05-2.asm

7. В файле lab5-2.asm заменил подпрограмму sprintf на sprint. Заново собрал исполняемый файл. Теперь вывод строки происходит без перехода на следующую строку.



```
mc [jsaidmagomedov@Ubuntu-VirtualBox]:~/work
/home/js~5-2.asm [ - - - - ] 11 L:[ 1+ 9 10/ 14]
%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 2.8: Программа lab05-2.asm

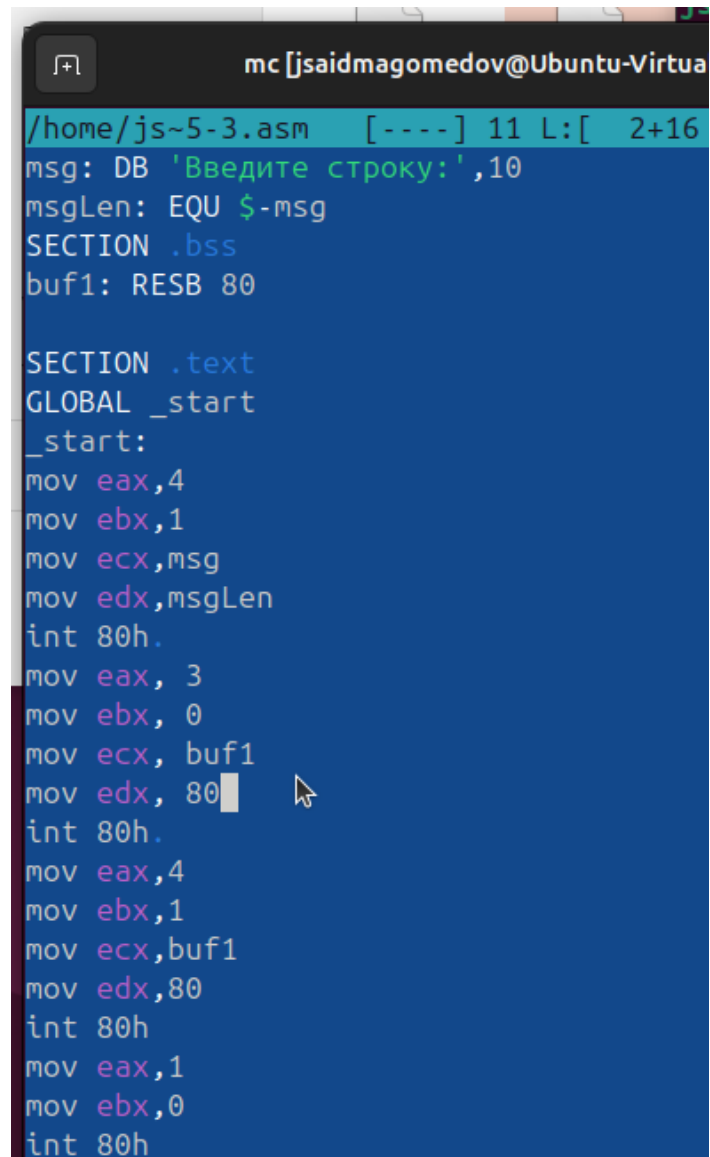


```
jsaidmagomedov@Ubuntu-VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab05-2.asm
jsaidmagomedov@Ubuntu-VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 lab05-2.o
-o lab05-2
jsaidmagomedov@Ubuntu-VirtualBox:~/work/arch-pc/lab05$ ./lab05-2
Введите строку: Jamal
jsaidmagomedov@Ubuntu-VirtualBox:~/work/arch-pc/lab05$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

Рисунок 2.9: Запуск программы lab05-2.asm

8. Скопировал программу lab05-1.asm и изменил код, чтобы вывести приглашение типа «Введите строку:», ввести строку с клавиатуры, вывести

введённую строку на экран.



```
mc [jsaidmagomedov@Ubuntu-Virtua
/home/js~5-3.asm [----] 11 L:[ 2+16
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

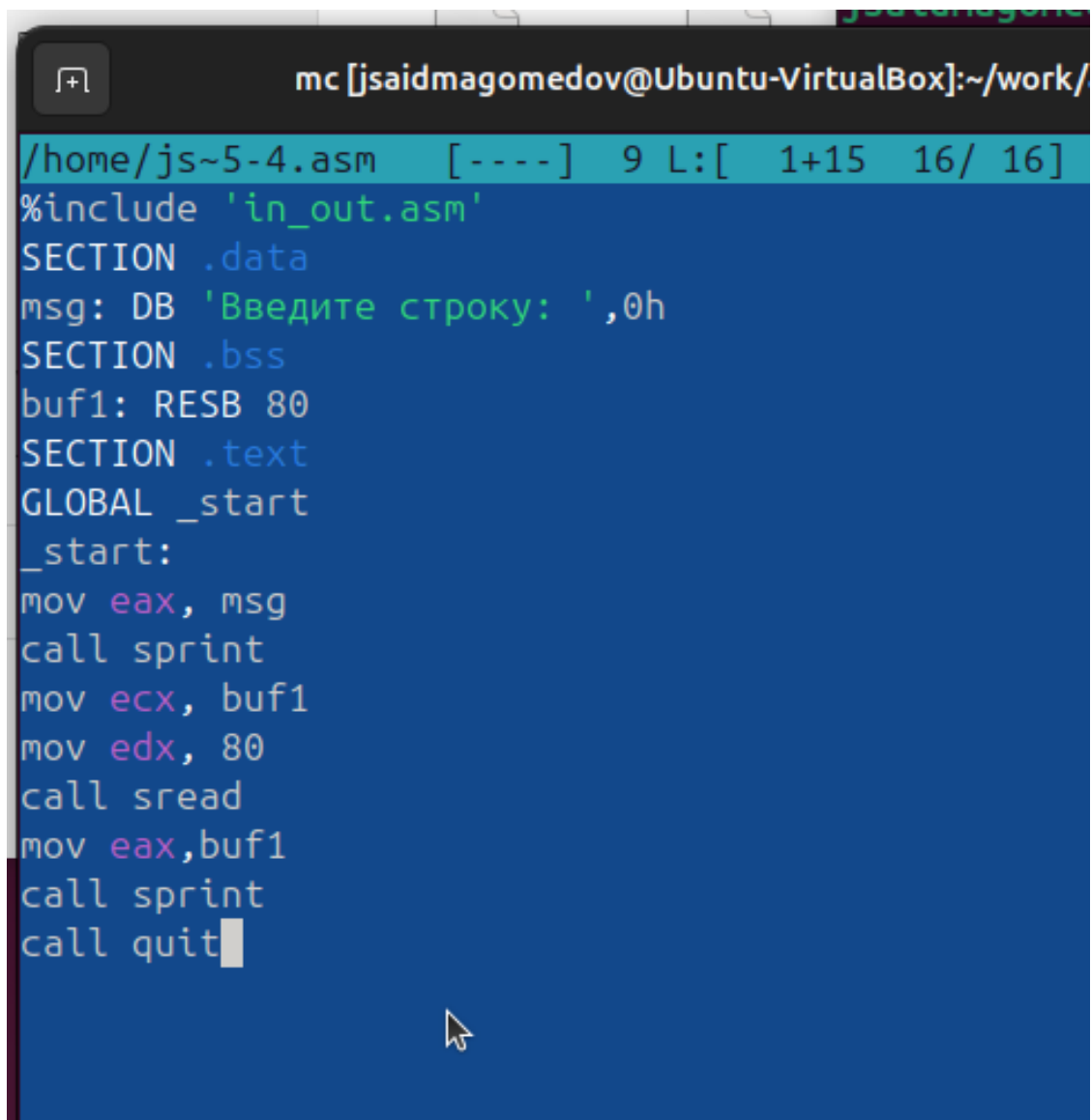
SECTION .text
GLOBAL _start
_start:
mov eax,4
mov ebx,1
mov ecx,msg
mov edx,msgLen
int 80h.
mov eax, 3
mov ebx, 0
mov ecx, buf1
mov edx, 80
int 80h.
mov eax,4
mov ebx,1
mov ecx,buf1
mov edx,80
int 80h
mov eax,1
mov ebx,0
int 80h
```

Рисунок 2.10: Программа lab05-3.asm

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

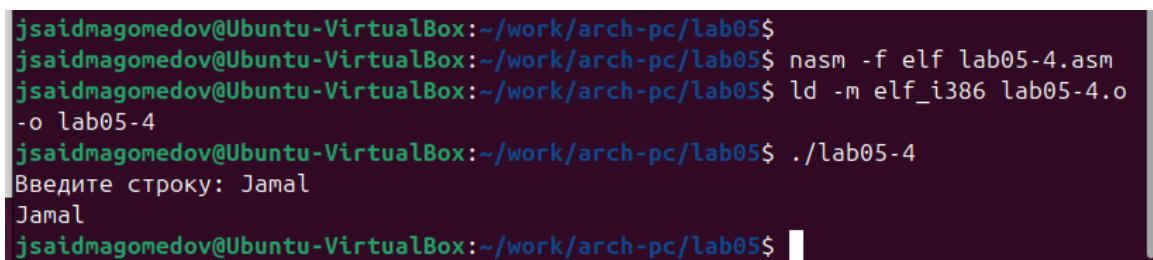
Рисунок 2.11: Запуск программы lab05-3.asm

15. Скопировал программу lab05-2.asm и сделал аналогично заданию выше, но теперь используются возможности из файла in_out.asm.



```
mc [jsaidmagomedov@Ubuntu-VirtualBox]:~/work/
/home/js~5-4.asm  [----]  9 L:[ 1+15 16/ 16]
%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 2.12: Программа lab05-4.asm



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

Рисунок 2.13: Запуск программы lab05-4.asm

3 Выводы

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