

Отчёт по лабораторной работе 5

дисциплина: Архитектура компьютера

Чернятин Артём Андреевич

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	16
3	Выводы	19

Список иллюстраций

2.1	Запуск Midnight Commander	6
2.2	Создание нового каталога	7
2.3	Создание файла lab05-1.asm	8
2.4	Выбор редактора mcedit	8
2.5	Написание программы lab05-1.asm	9
2.6	Просмотр кода lab05-1.asm	10
2.7	Тестовый запуск lab05-1.asm	10
2.8	Перемещение файла in_out.asm	11
2.9	Создание копии файла lab05-1.asm	12
2.10	Добавление подпрограмм в lab05-2.asm	13
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	16
2.15	Тестовый запуск lab05-3.asm	17
2.16	Обновлённая программа lab05-4.asm	17
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-pc. Затем создаю новый каталог под названием lab05, нажав F7 (см. рис. 2.2).

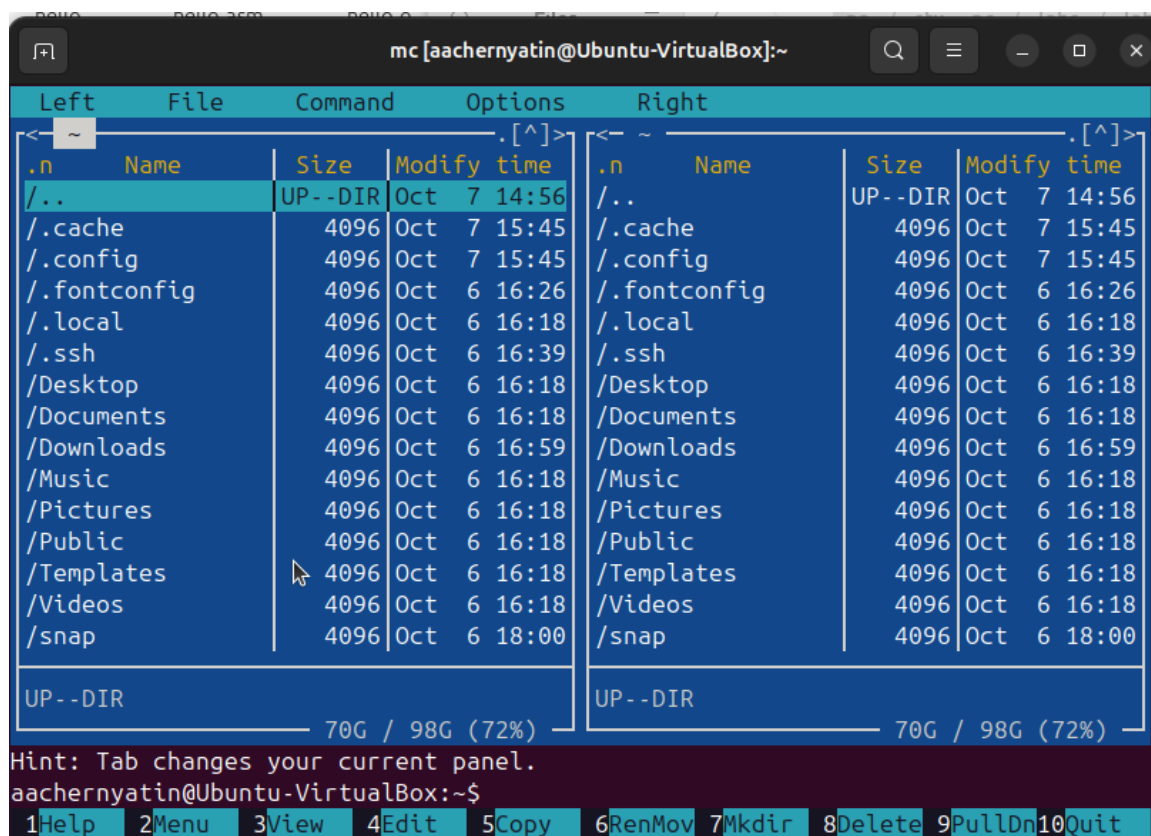


Рисунок 2.1: Запуск Midnight Commander

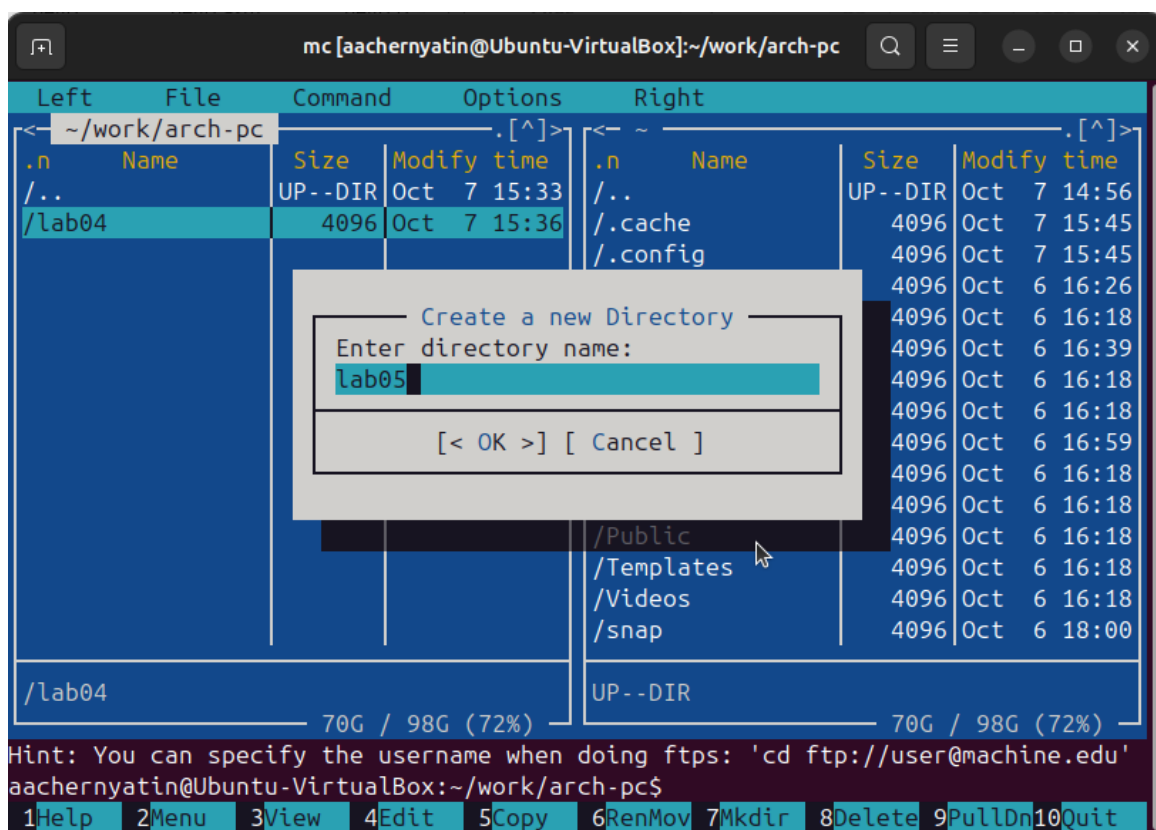
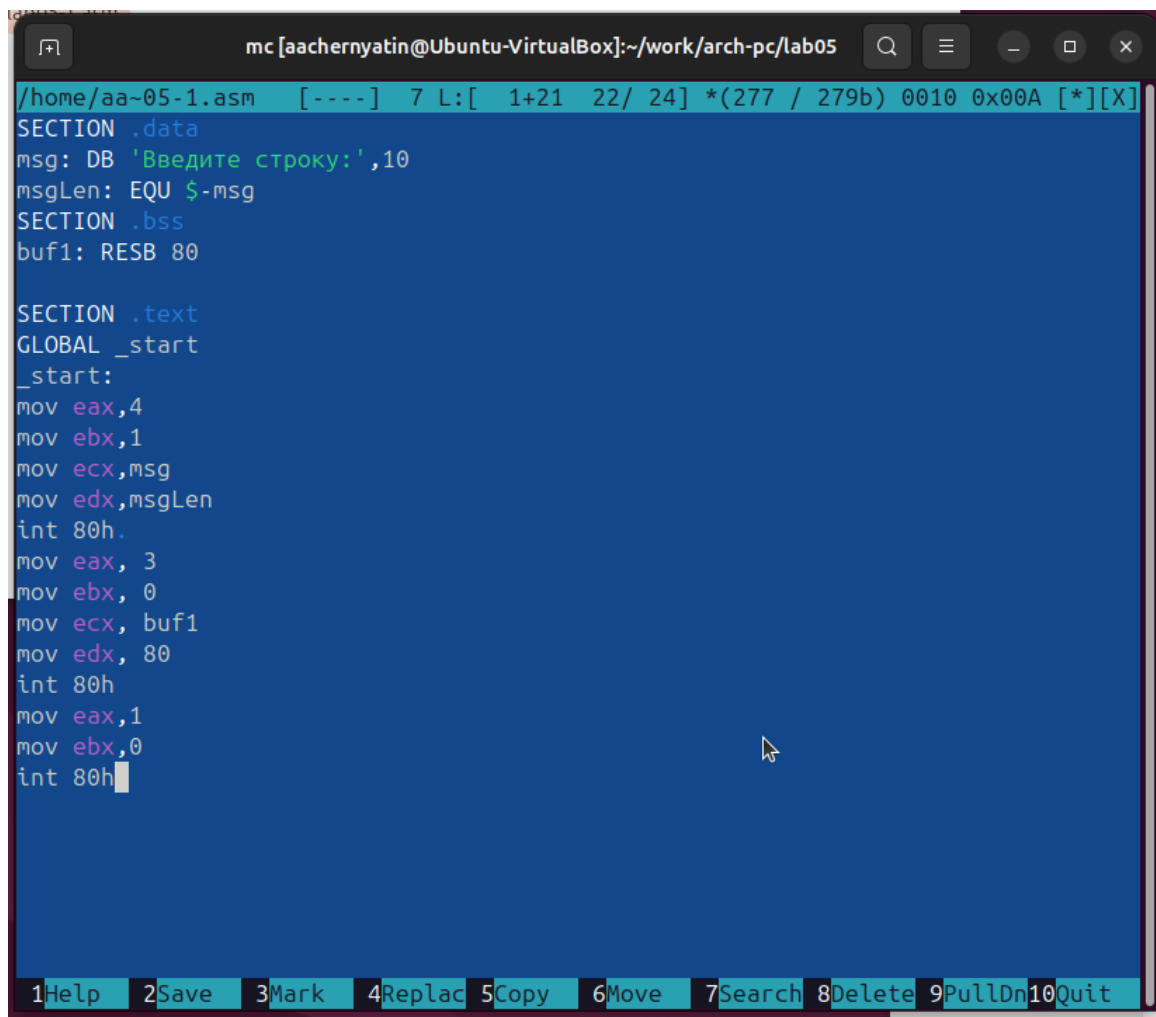


Рисунок 2.2: Создание нового каталога

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

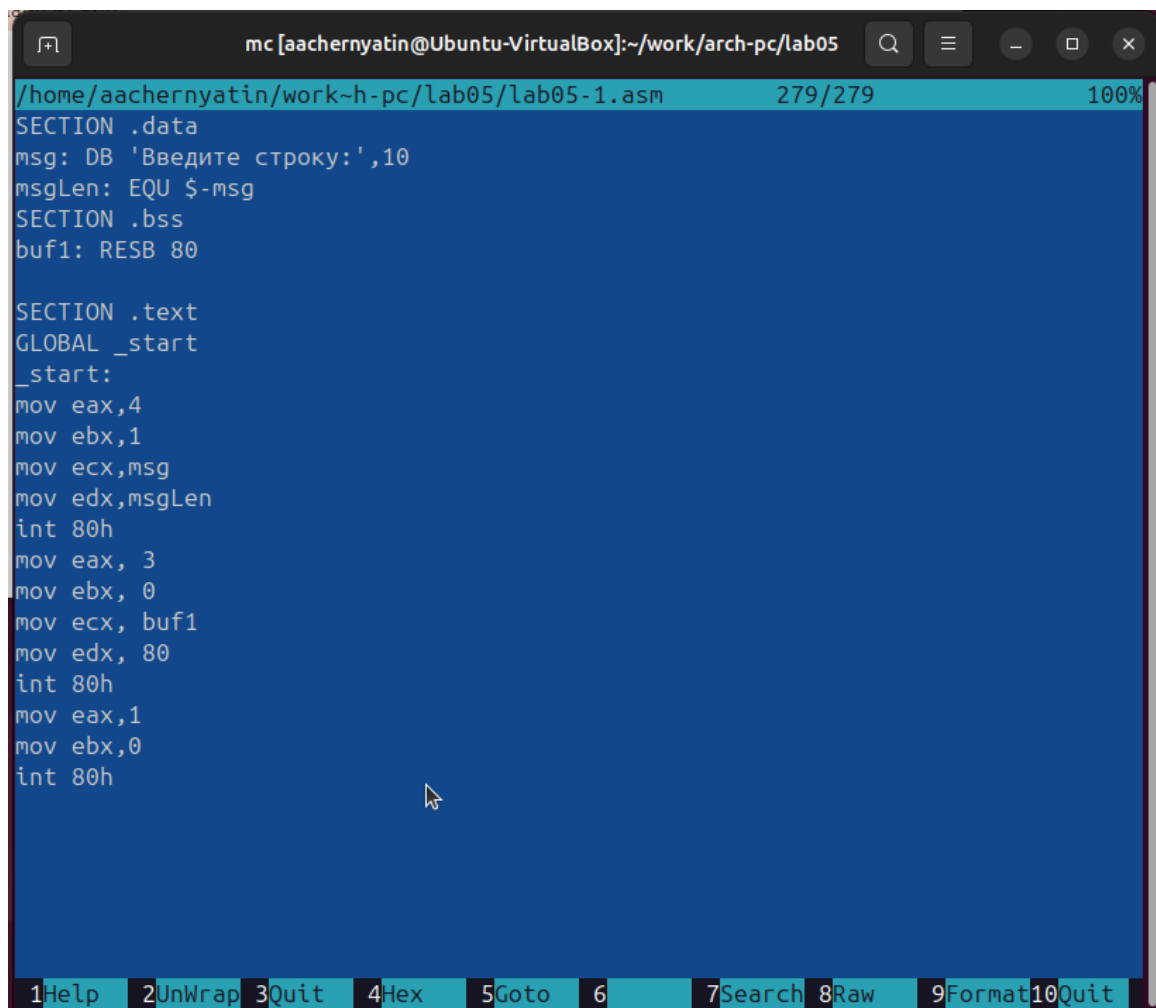


```
mc [aachernyatin@Ubuntu-VirtualBox]:~/work/arch-pc/lab05
/home/aa~05-1.asm [----] 7 L:[ 1+21 22/ 24] *(277 / 279b) 0010 0x00A [*][X]
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

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
1Help 2Save 3Mark 4Replac 5Copy 6Move 7Search 8Delete 9PullDn10Quit
```

Рисунок 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).



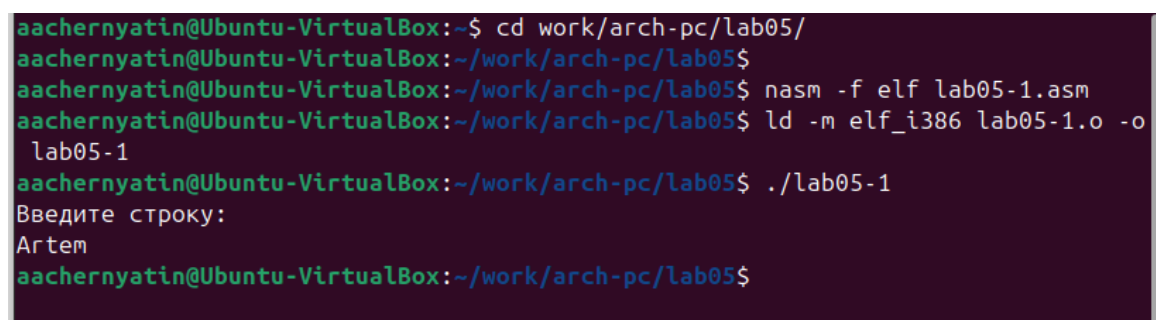
```
mc [aachernyatin@Ubuntu-VirtualBox]:~/work/arch-pc/lab05
/home/aachernyatin/work-h-pc/lab05/lab05-1.asm 279/279 100%
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg
SECTION .bss
buf1: RESB 80

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

1Help 2UnWrap 3Quit 4Hex 5Goto 6 7Search 8Raw 9Format10Quit
```

Рисунок 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).



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

Рисунок 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.

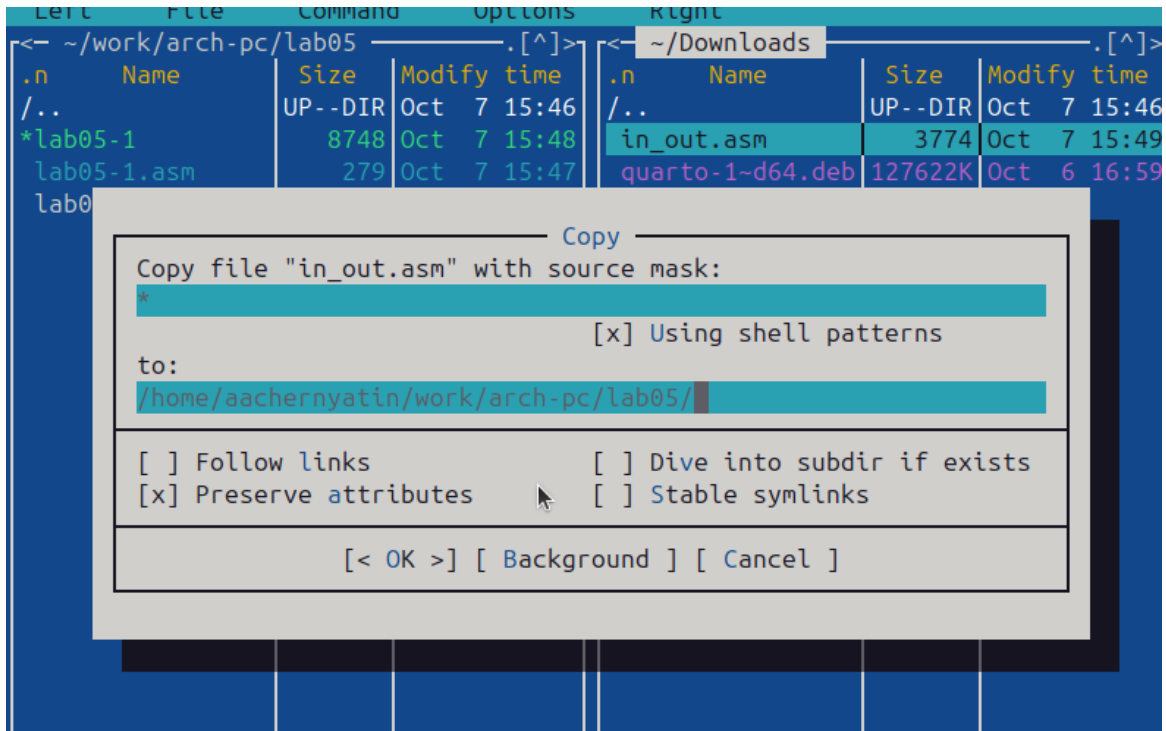


Рисунок 2.8: Перемещение файла in_out.asm

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

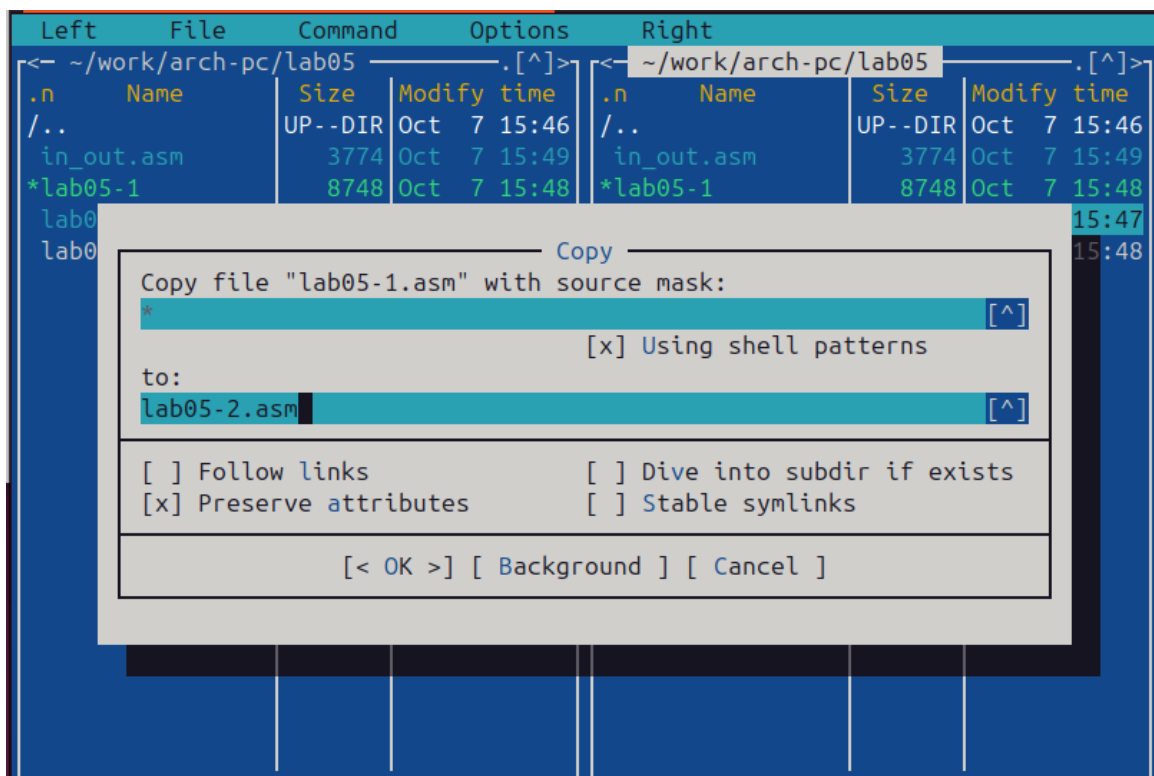
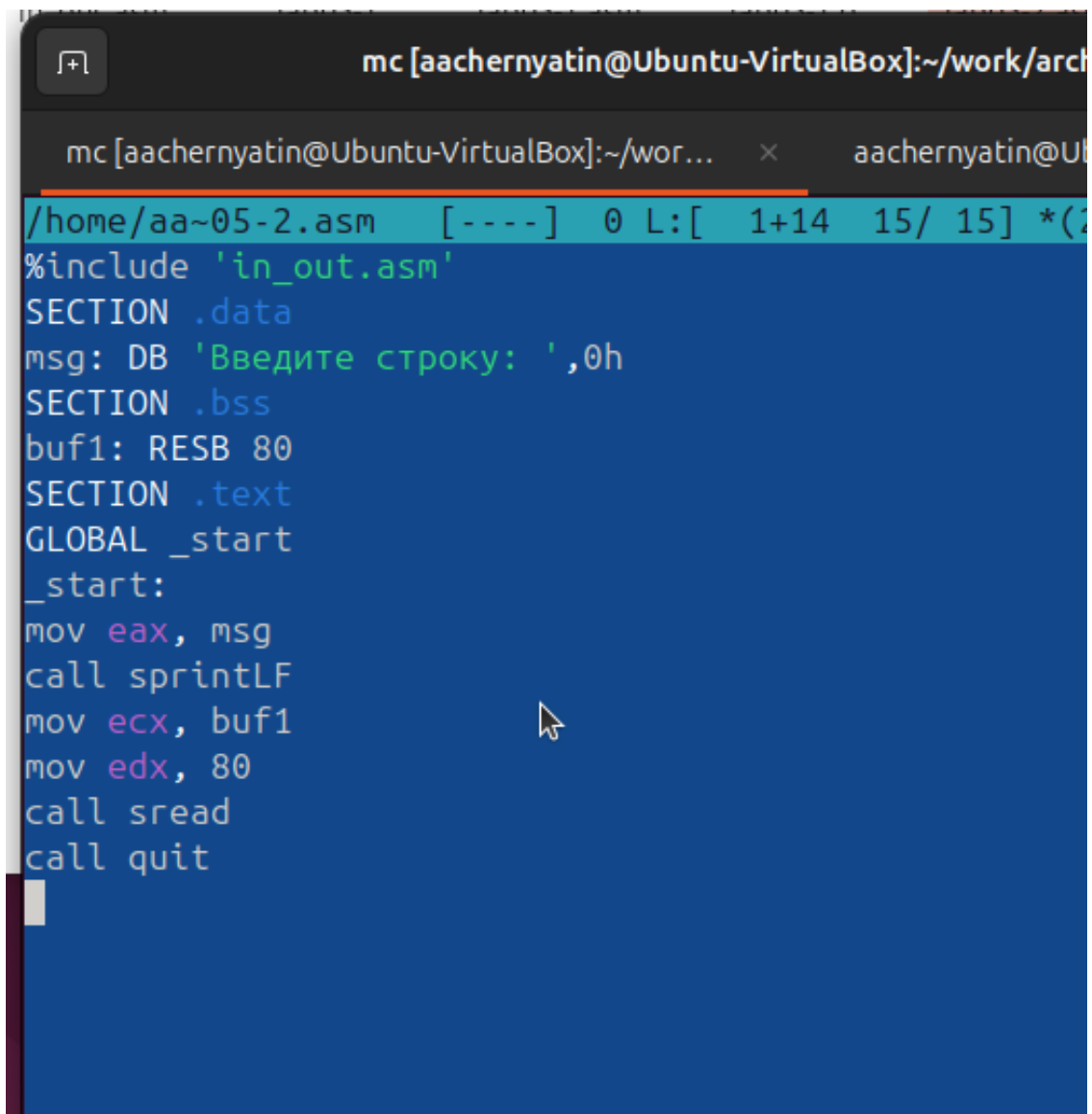


Рисунок 2.9: Создание копии файла lab05-1.asm

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



```
mc [aachernyatin@Ubuntu-VirtualBox]:~/work/arch
mc [aachernyatin@Ubuntu-VirtualBox]:~/wor... × aachernyatin@U
/home/aa~05-2.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 sprintLF
mov ecx, buf1
mov edx, 80
call sread
call quit
```

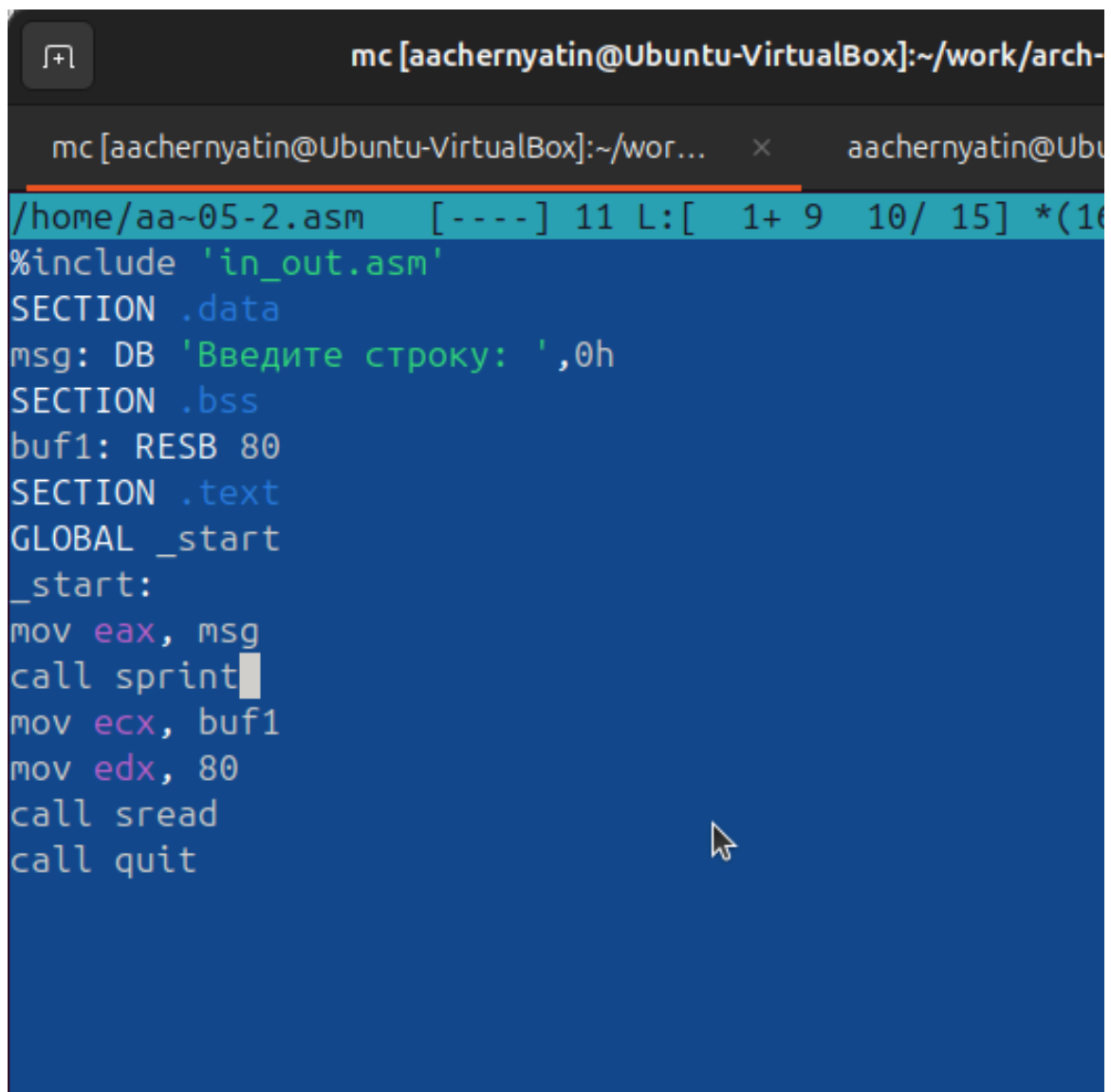
Рисунок 2.10: Добавление подпрограмм в lab05-2.asm

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

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

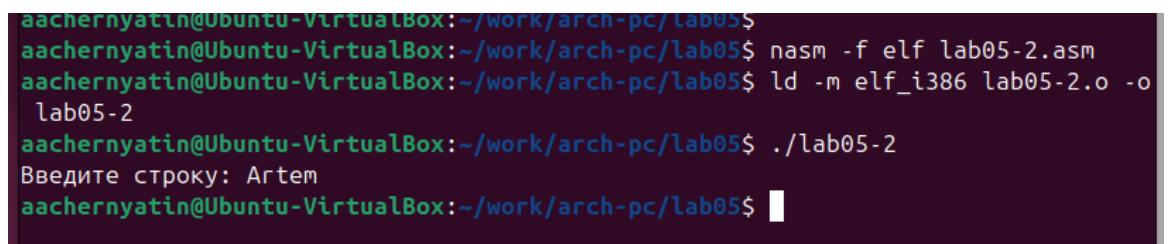
Рисунок 2.11: Запуск программы lab05-2.asm

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



```
mc [aachernyatin@Ubuntu-VirtualBox]:~/work/arch-  
mc [aachernyatin@Ubuntu-VirtualBox]:~/wor... × aachernyatin@Ubu  
/home/aa~05-2.asm [----] 11 L:[ 1+ 9 10/ 15] *(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  
call quit
```

Рисунок 2.12: Обновлённая версия lab05-2.asm



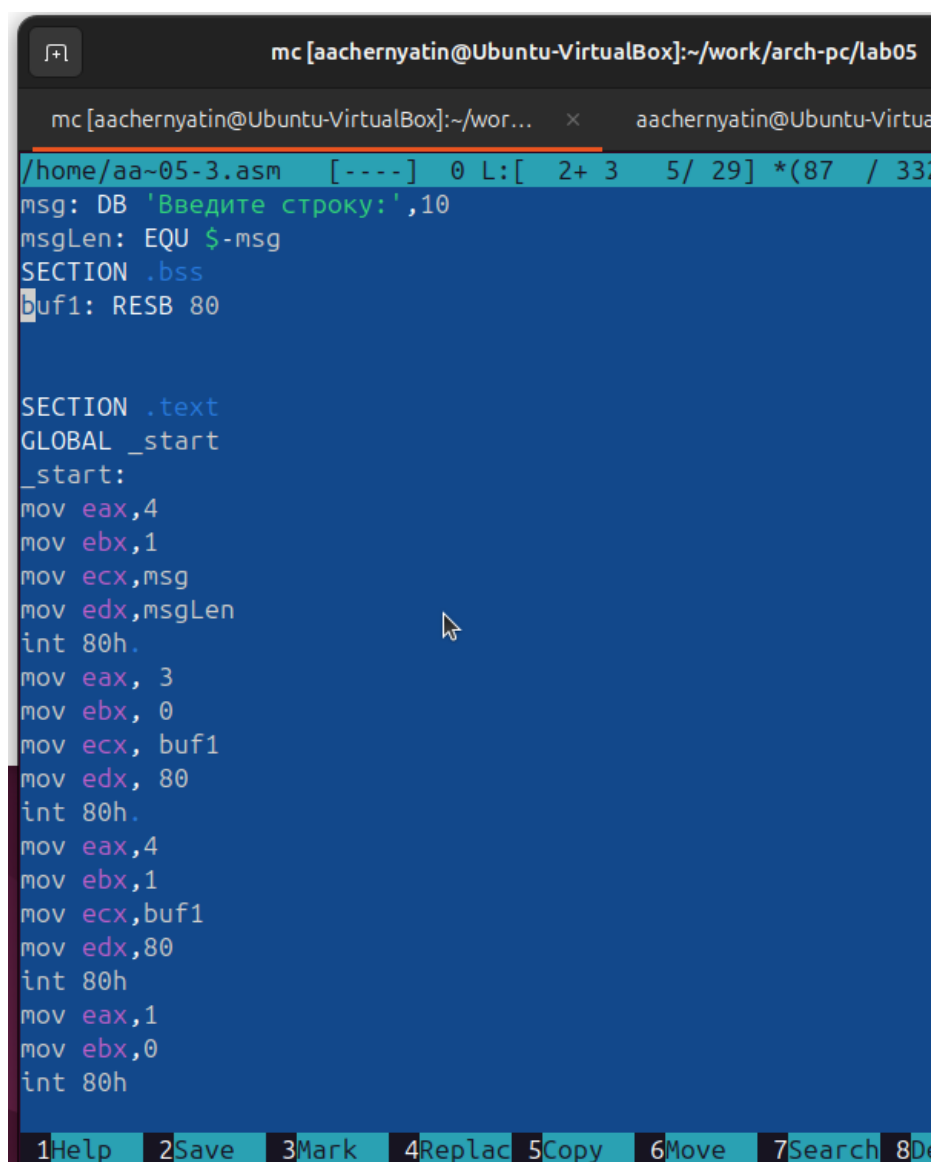
```
aachernyatin@Ubuntu-VirtualBox:~/work/arch-pc/lab05$  
aachernyatin@Ubuntu-VirtualBox:~/work/arch-pc/lab05$ nasm -f elf lab05-2.asm  
aachernyatin@Ubuntu-VirtualBox:~/work/arch-pc/lab05$ ld -m elf_i386 lab05-2.o -o  
lab05-2  
aachernyatin@Ubuntu-VirtualBox:~/work/arch-pc/lab05$ ./lab05-2  
Введите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

Рисунок 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.



```
mc [aachernyatin@Ubuntu-VirtualBox]:~/work/arch-pc/lab05
/home/aa~05-3.asm [----] 0 L:[ 2+ 3 5/ 29] *(87 / 332
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 2.14: Редактирование программы lab05-3.asm

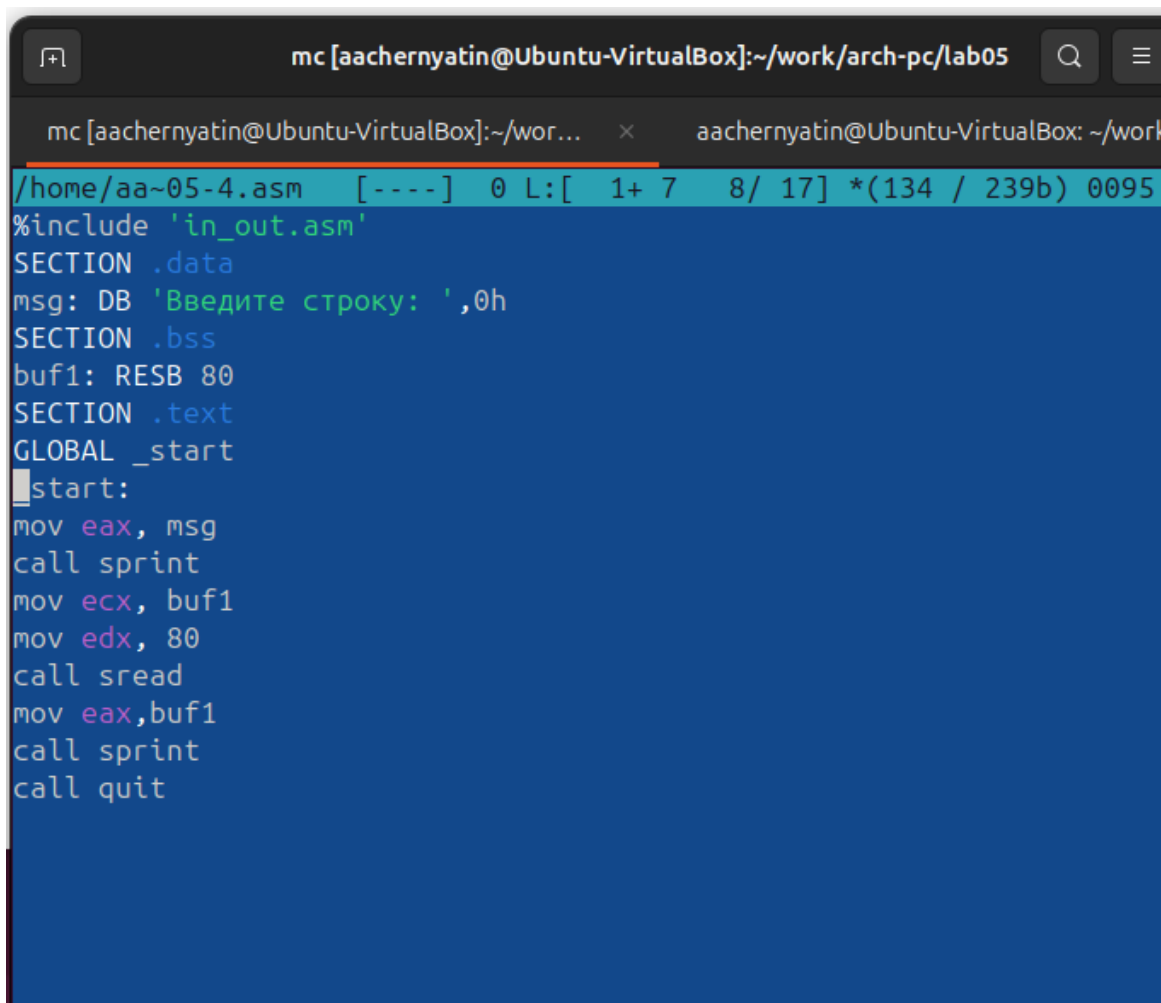

```

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

```

Рисунок 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).



```

mc [aachernyatin@Ubuntu-VirtualBox]:~/work/arch-pc/lab05
aachernyatin@Ubuntu-VirtualBox: ~/work
/home/aa~05-4.asm  [----]  0 L:[ 1+ 7 8/ 17] *(134 / 239b) 0095
%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 2.16: Обновлённая программа lab05-4.asm

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

Рисунок 2.17: Тестовый запуск lab05-4.asm

3 Выводы

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