

# **Отчёт по лабораторной работе 5**

**Основы работы с Midnight Commander. Структура программы на языке ассемблера NASM. Системные вызовы в ОС GNU Linux**

Вахиш Дуаа Иссам Али Ахмед

# **Содержание**

<b>1 Цель работы</b>	<b>5</b>
<b>2 Выполнение лабораторной работы</b>	<b>6</b>
2.1 Знакомство с Midnight Commander . . . . .	6
2.2 Подключение внешнего файла in_out.asm . . . . .	10
2.3 Задание для самостоятельной работы . . . . .	14
<b>3 Выводы</b>	<b>18</b>

# Список иллюстраций

2.1 Запуск Midnight Commander . . . . .	6
2.2 Создание каталога lab05 . . . . .	7
2.3 Создание файла lab05-1.asm . . . . .	7
2.4 Редактирование программы в файле lab05-1.asm . . . . .	8
2.5 Просмотр содержимого файла lab05-1.asm . . . . .	9
2.6 Запуск программы lab05-1.asm . . . . .	10
2.7 Копирование файла in_out.asm в рабочий каталог . . . . .	10
2.8 Копирование файла lab05-1.asm в lab05-2.asm . . . . .	11
2.9 Программа в файле lab05-2.asm . . . . .	12
2.10 Запуск программы lab05-2.asm . . . . .	12
2.11 Обновленная программа в lab05-2.asm . . . . .	13
2.12 Запуск обновленной программы lab05-2.asm . . . . .	13
2.13 Копирование файла lab05-1.asm для нового задания . . . . .	14
2.14 Код программы в файле lab05-3.asm . . . . .	15
2.15 Запуск программы lab05-3.asm . . . . .	15
2.16 Копирование файла lab05-2.asm для модификации . . . . .	16
2.17 Код программы в файле lab05-4.asm . . . . .	17
2.18 Запуск программы lab05-4.asm . . . . .	17

# **Список таблиц**

# **1 Цель работы**

Целью работы является приобретение практических навыков работы в Midnight Commander, а также освоение инструкций языка ассемблера mov и int.

## 2 Выполнение лабораторной работы

### 2.1 Знакомство с Midnight Commander

Я запустила Midnight Commander и, используя стрелочные клавиши и клавишу Enter, перешла в каталог ~/work/arch-pc. Затем нажала F7 для создания нового каталога lab05.

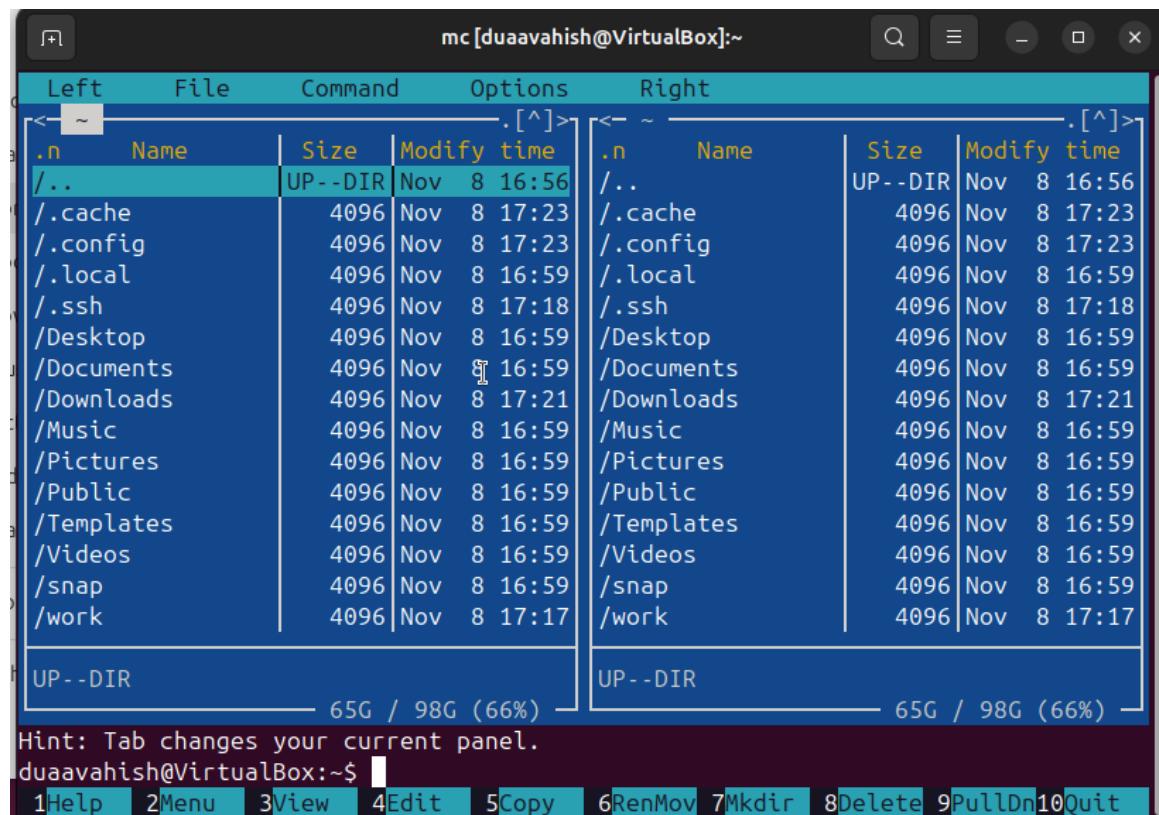


Рисунок 2.1: Запуск Midnight Commander

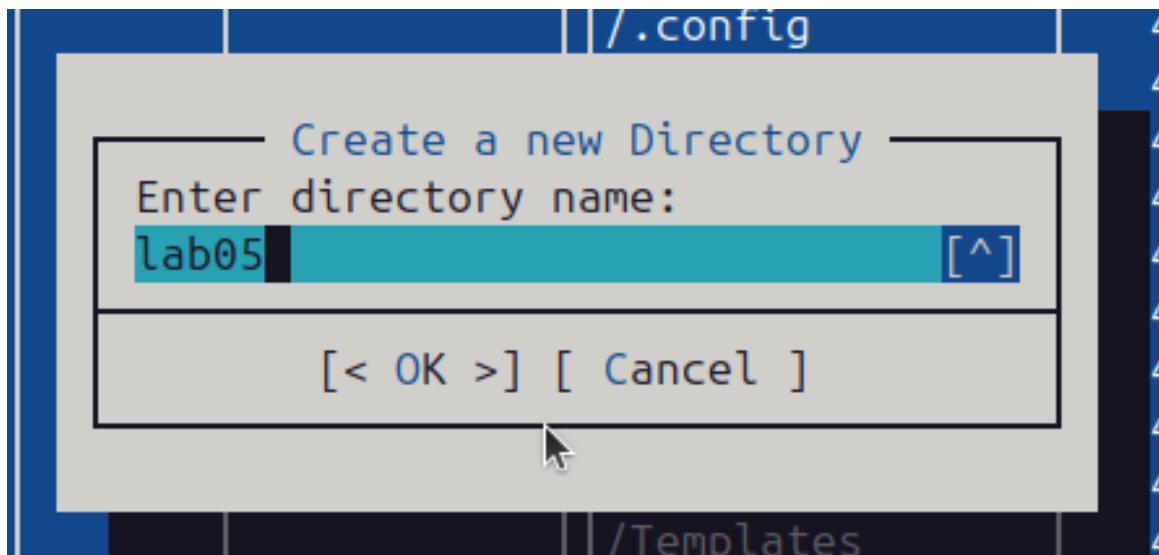


Рисунок 2.2: Создание каталога lab05

С помощью команды touch я создала файл lab05-1.asm.

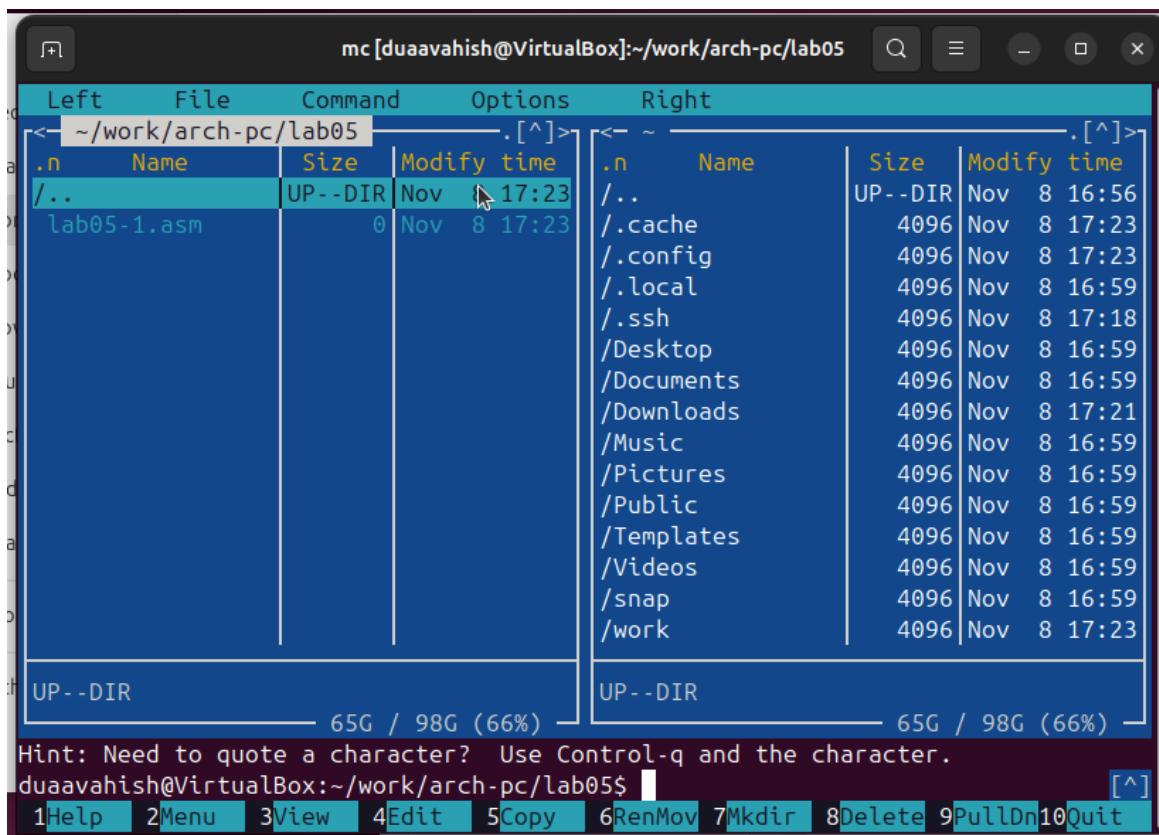
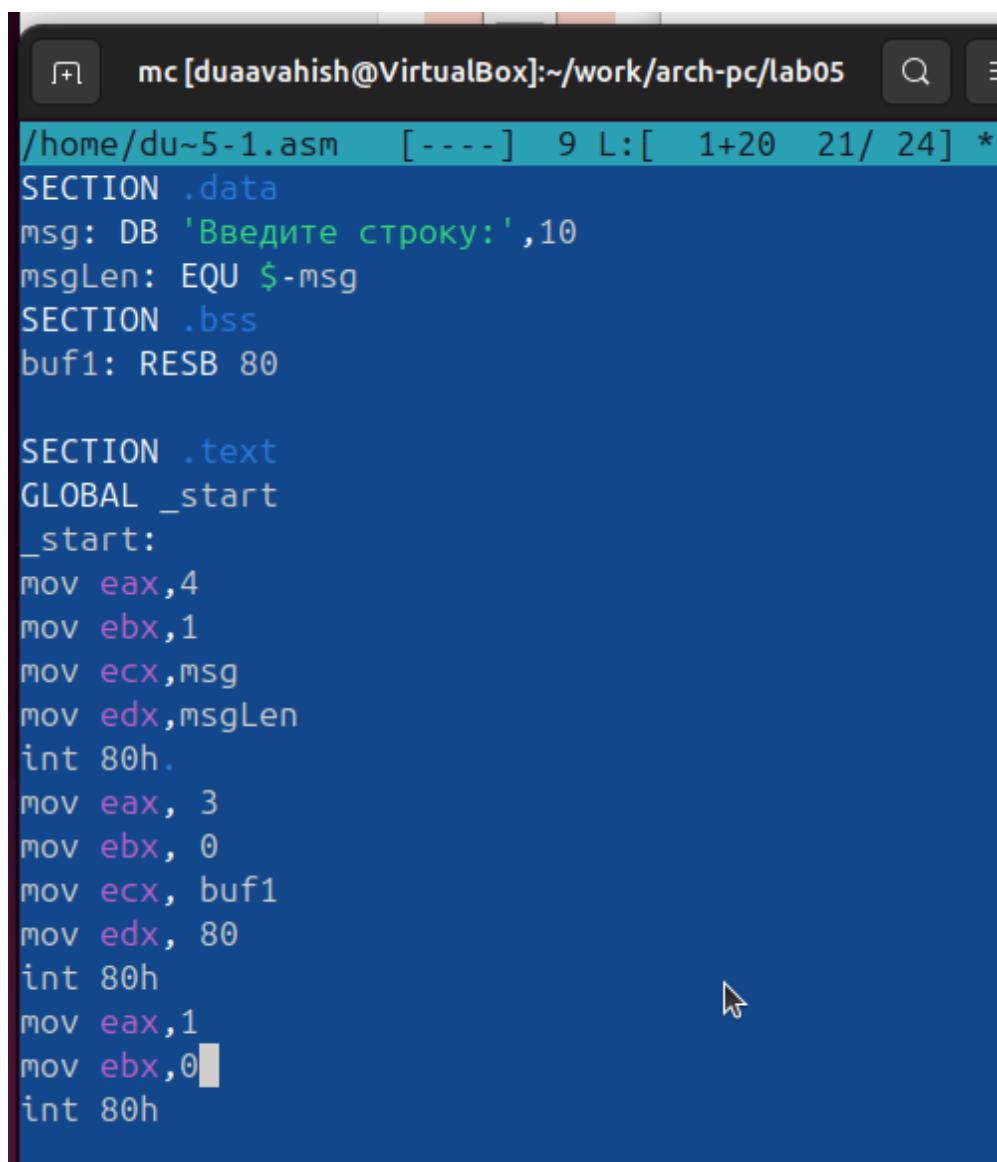


Рисунок 2.3: Создание файла lab05-1.asm

Я открыла файл для редактирования, нажав F4, выбрала редактор mceditor и написала код программы в соответствии с заданием.



The screenshot shows a terminal window titled 'mc [duaavahish@VirtualBox]:' with the command '/home/du~5-1.asm' entered. The file content is an assembly program:

```
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: Редактирование программы в файле lab05-1.asm

Для проверки я открыла файл на просмотр, нажав F3, и убедилась, что он содержит правильный код.



The screenshot shows the Mars Cross Assembler (mc) interface. The title bar reads "mc [duaavahish@VirtualBox]:~/work/arch-pc/lab05". The status bar at the bottom indicates the file path "/home/duaavahish/05/lab05-1.asm" and the line count "279/279". The main window displays the assembly code:

```
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 содержимого файла lab05-1.asm

Я скомпилировала программу, сгенерировала объектный файл, произвела компоновку и запустила исполняемый файл, чтобы убедиться в корректности работы.

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

Рисунок 2.6: Запуск программы lab05-1.asm

## 2.2 Подключение внешнего файла in\_out.asm

Я скачала файл in\_out.asm и разместила его в рабочем каталоге. Для копирования я использовала клавишу F5, а для перемещения – клавишу F6.

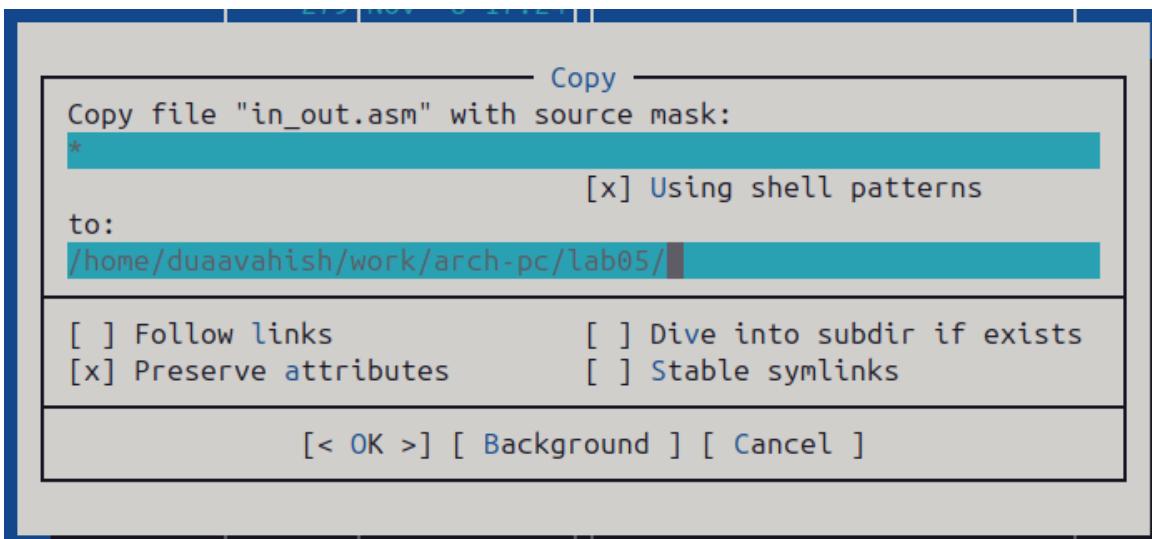


Рисунок 2.7: Копирование файла in\_out.asm в рабочий каталог

Я также скопировала файл lab05-1.asm в новый файл lab05-2.asm.

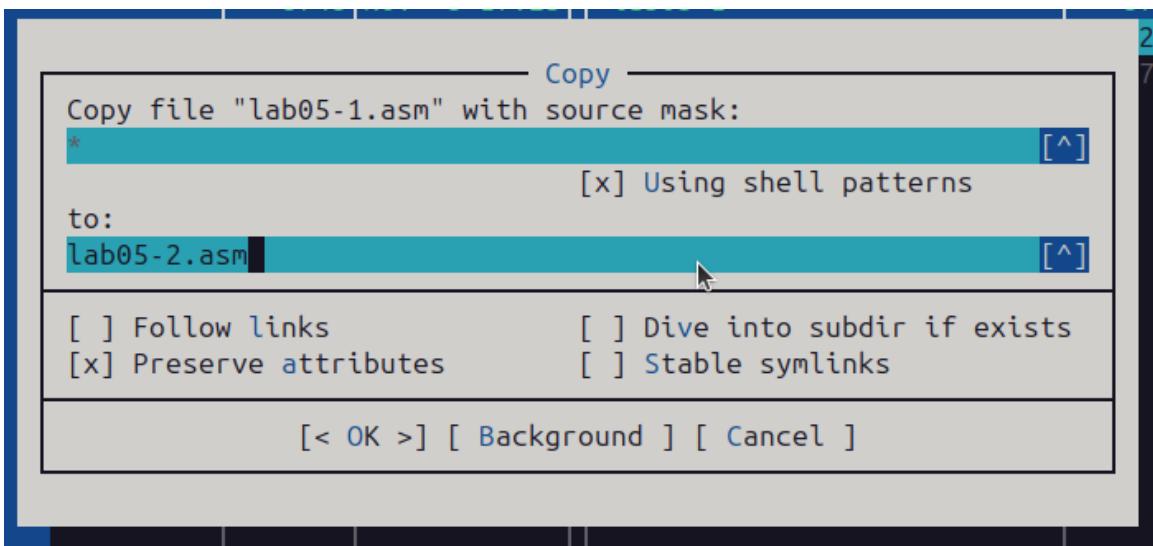


Рисунок 2.8: Копирование файла lab05-1.asm в lab05-2.asm

В файле lab05-2.asm я написала код программы, используя подпрограммы из внешнего файла in\_out.asm. Программа была успешно скомпилирована и протестирована.

The screenshot shows a terminal window with a dark blue background. In the top right corner, the text "mc [duaawahish@VirtualBox ~]" is visible. The main area of the terminal contains assembly code:

```
/home/duaawahish/wo~c/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
```

Рисунок 2.9: Программа в файле lab05-2.asm

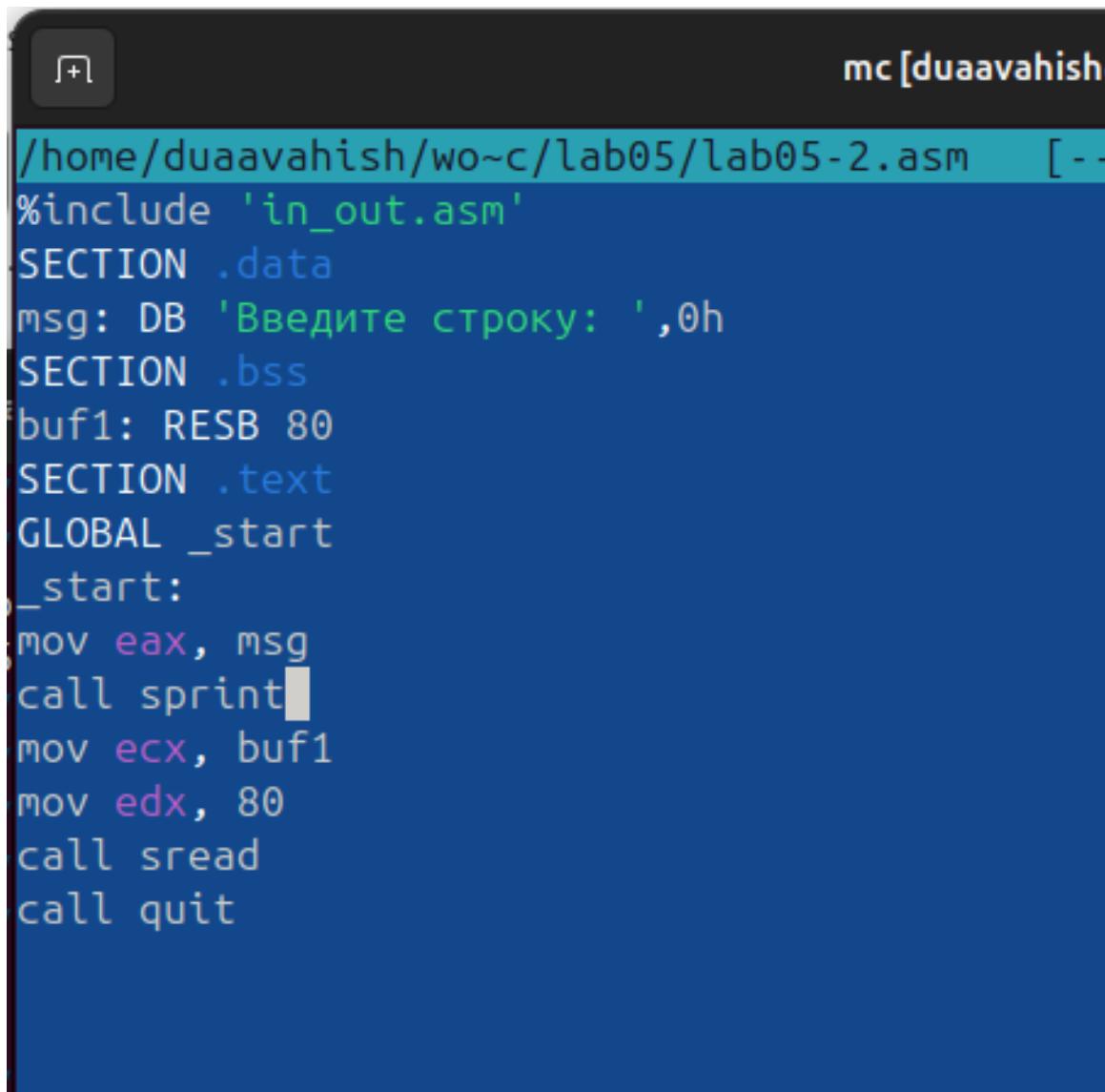
The screenshot shows a terminal window with a dark blue background. The text in the window is:

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

Рисунок 2.10: Запуск программы lab05-2.asm

В файле lab05-2.asm я заменила подпрограмму sprintLF на sprint. После пе-

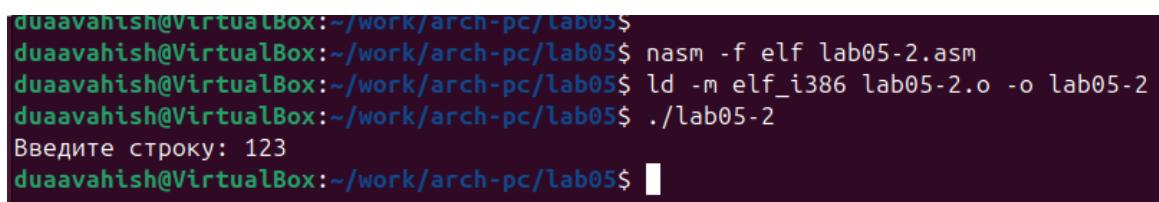
ресурсами исполняемого файла, теперь вывод строки не завершается символом новой строки.



The screenshot shows a terminal window with the title 'mc [duaavahish]'. The code in the editor is as follows:

```
/home/duaavahish/wo~c/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 sprint
    mov ecx, buf1
    mov edx, 80
    call sread
    call quit
```

Рисунок 2.11: Обновленная программа в lab05-2.asm



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

Рисунок 2.12: Запуск обновленной программы lab05-2.asm

## 2.3 Задание для самостоятельной работы

Я скопировала программу lab05-1.asm и изменила код так, чтобы он работал по следующему алгоритму:

- вывести приглашение с текстом «Введите строку:»;
- считать строку с клавиатуры;
- вывести введённую строку на экран.

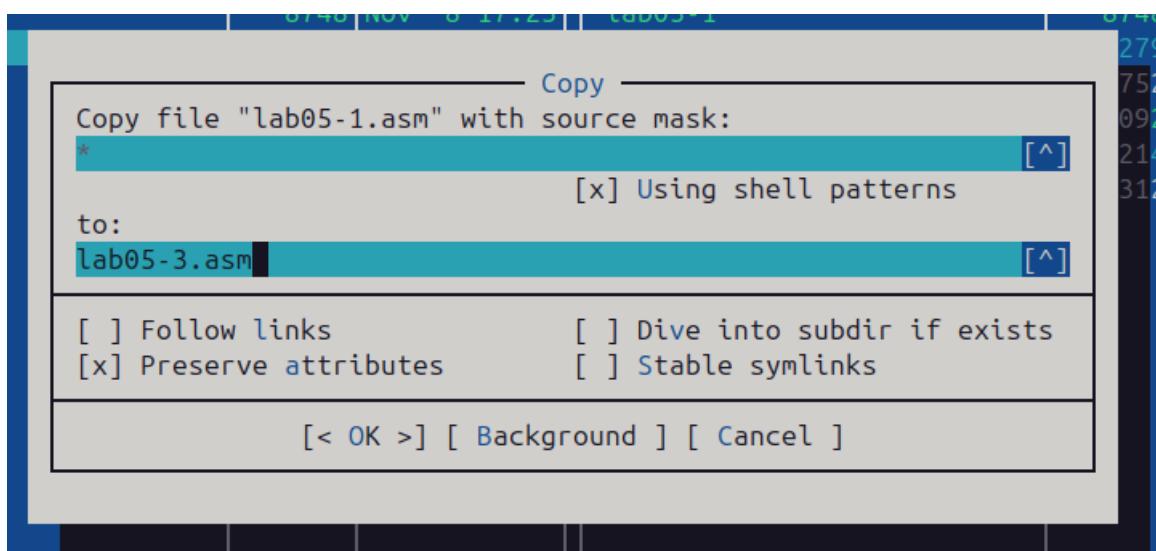


Рисунок 2.13: Копирование файла lab05-1.asm для нового задания

The screenshot shows a terminal window titled 'mc [duaavahish]' displaying assembly code. The code defines a message in the .data section, allocates memory in the .bss section, and implements a read loop in the .text section using system calls. The assembly code uses standard conventions for registers (eax, ebx, ecx, edx) and memory addresses.

```
//home/duaavahish/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
```

Рисунок 2.14: Код программы в файле lab05-3.asm

The screenshot shows a terminal window with the following session:

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

Рисунок 2.15: Запуск программы lab05-3.asm

Аналогично, я скопировала программу lab05-2.asm и внесла изменения в код, теперь используя подпрограммы из файла in\_out.asm.

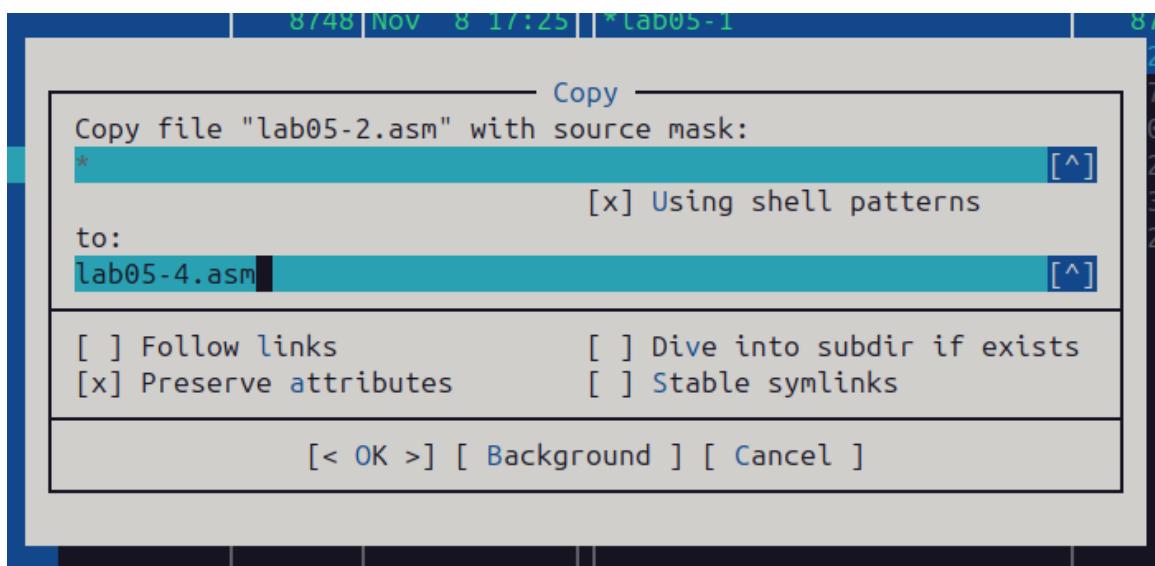


Рисунок 2.16: Копирование файла lab05-2.asm для модификации

The screenshot shows a terminal window with a dark blue background. In the top right corner, it says "mc [duaav". The main area contains assembly code:

```
/home/duaavahish/work~pc/lab05/lab05-4.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 sprint
    mov ecx, buf1
    mov edx, 80
    call sread
    mov eax, buf1
    call sprint
    call quit
```

Рисунок 2.17: Код программы в файле lab05-4.asm

The screenshot shows a terminal window with a dark blue background. The user runs the assembly program through the command line:

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

Рисунок 2.18: Запуск программы lab05-4.asm

## **3 Выводы**

Я научилась писать базовые ассемблерные программы и освоила ассемблерные инструкции mov и int.