

# **Отчет по лабораторной работе № 6**

**Нкнбд-05-2023**

Диого Элизеу Луиж Музумбо

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>4</b>
<b>2</b>	<b>Задание</b>	<b>5</b>
<b>3</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
3.1	Работа с Midnight Commander . . . . .	6
3.2	Задание для самостоятельной работы . . . . .	12
<b>4</b>	<b>Выводы</b>	<b>15</b>

# Список иллюстраций

3.1	Midnight Commander . . . . .	6
3.2	Каталог ~/work/arch-pc . . . . .	6
3.3	lab06 . . . . .	7
3.4	touch . . . . .	7
3.5	lab6-1.asm . . . . .	7
3.6	Файл lab6-1.asm для редактирования . . . . .	8
3.7	Текст программы из листинга 6.1 . . . . .	8
3.8	Файл содержащий текст программы . . . . .	9
3.9	Получившийся исполняемый файл . . . . .	9
3.10	Скачанный файл in_out.asm . . . . .	10
3.11	Копирование файла in_out.asm . . . . .	10
3.12	lab6-2.asm . . . . .	11
3.13	sprintLF, sread и quit . . . . .	11
3.14	Исполняемый файл . . . . .	11
3.15	sprint . . . . .	12
3.16	Перенос строки перед вводом текста . . . . .	12
3.17	lab6-3.asm . . . . .	12
3.18	Изменения в программе . . . . .	13
3.19	Проверка работы файла . . . . .	13
3.20	lab6-4.asm . . . . .	14
3.21	Использование подпрограмм из внешнего файла in_out.asm . . . . .	14
3.22	Проверка работы исполняемого файла . . . . .	14

# 1 Цель работы

Приобретение практических навыков работы в Midnight Commander. Освоение инструкций языка ассемблера `mov` и `int`.

## 2 Задание

Создать исполняющие файлы из файлов ассемблера, реализующие ввод и вывод текста.

## 3 Выполнение лабораторной работы

### 3.1 Работа с Midnight Commander

Открыли Midnight Commander (рис. 3.1)

```
[elmdiago@fedora ~]$ mc
```

Рис. 3.1: Midnight Commander

Пользуясь клавишами **←**, **→** и **Enter** перешли в каталог `~/work/arch-pc`, созданный при выполнении лабораторной работы №5. (рис. 3.2)

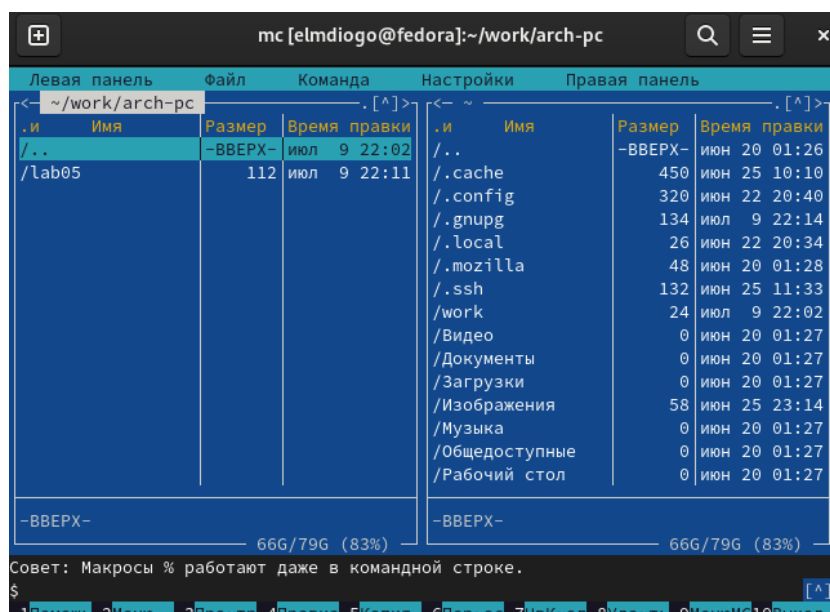


Рис. 3.2: Каталог `~/work/arch-pc`

С помощью функциональной клавиши F7 создали папку lab06 и перешли в созданный каталог. (рис. 3.3)

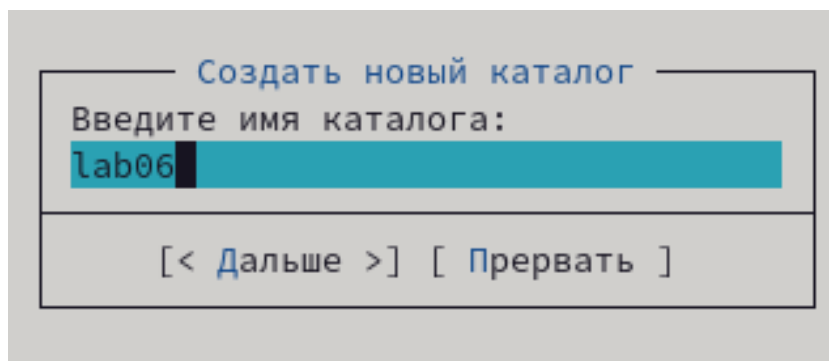


Рис. 3.3: lab06

Пользуясь строкой ввода и командой touch создайте файл lab6-1.asm. (рис. 3.4), (рис. 3.5)

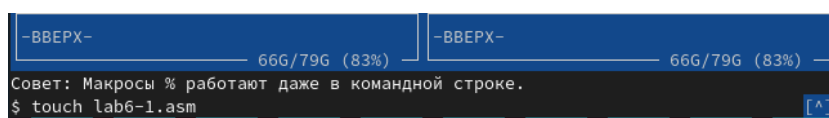


Рис. 3.4: touch

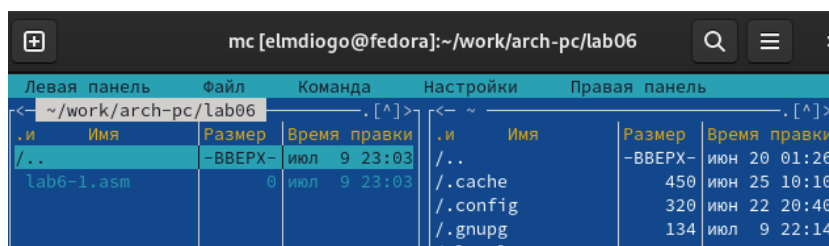


Рис. 3.5: lab6-1.asm

С помощью функциональной клавиши F4 откройте файл lab6-1.asm для редактирования во встроенном редакторе. (рис. 3.6)

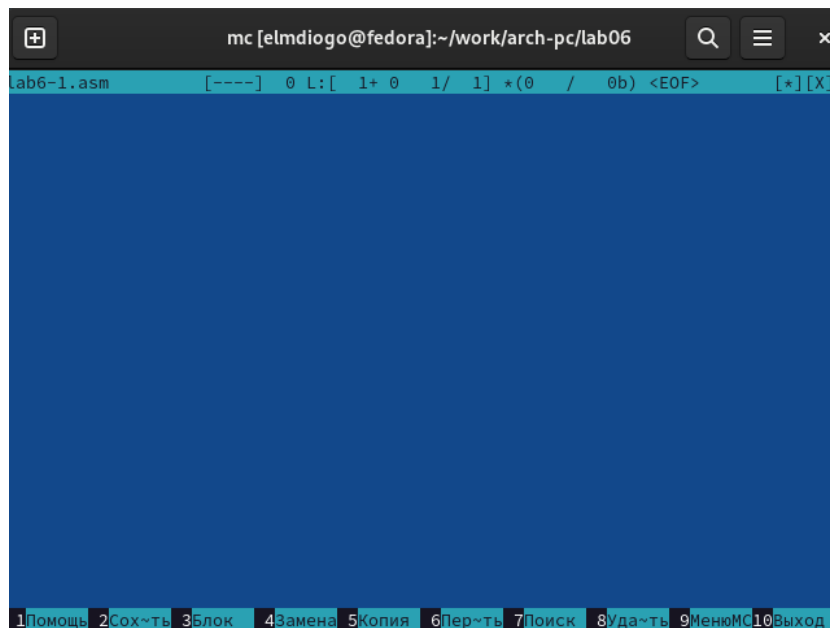


Рис. 3.6: Файл lab6-1.asm для редактирования

Ввели текст программы из листинга 6.1, сохранили изменения и закрыли файл.  
(рис. 3.7)

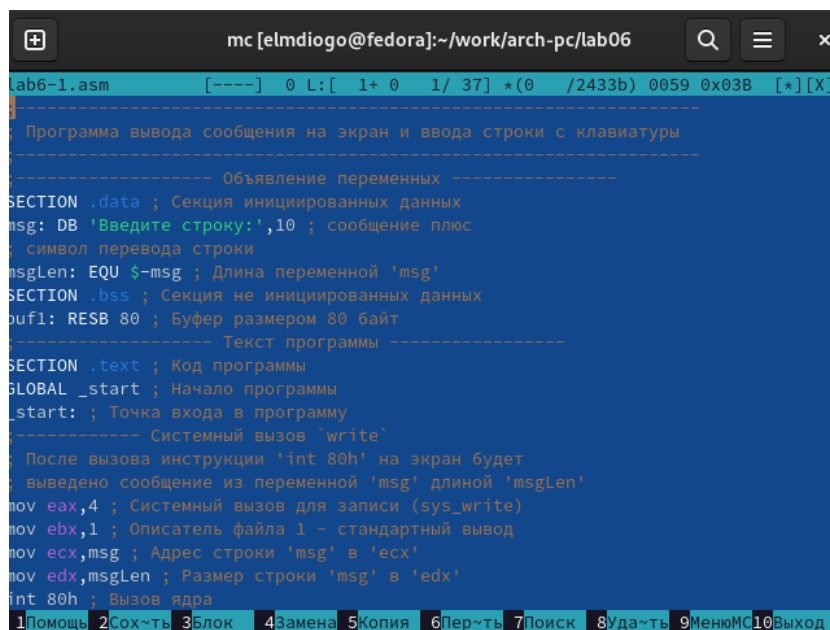
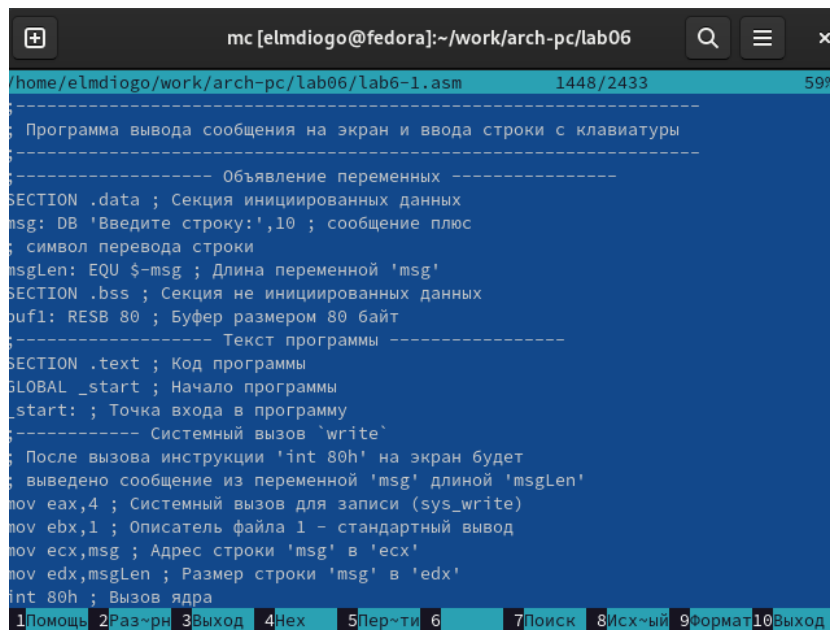


Рис. 3.7: Текст программы из листинга 6.1

С помощью функциональной клавиши F3 открыли файл lab6-1.asm для про-



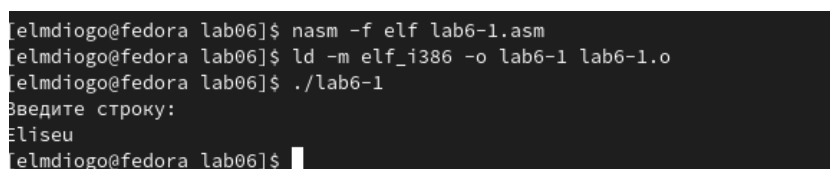
смотра. Убедились, что файл содержит текст программы. (рис. 3.8)



```
mc [elmdiago@fedora]:~/work/arch-pc/lab06
/home/elmdiago/work/arch-pc/lab06/lab6-1.asm 1448/2433 59%
-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
-----
;----- Объявление переменных -----
SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
; символ перевода строки
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
;----- Текст программы -----
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
;----- Системный вызов `write`
; После вызова инструкции `int 80h` на экран будет
; выведено сообщение из переменной 'msg' длиной 'msgLen'
mov eax,4 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Описатель файла 1 - стандартный вывод
mov ecx,msg ; Адрес строки 'msg' в 'ecx'
mov edx,msgLen ; Размер строки 'msg' в 'edx'
int 80h ; Вызов ядра
1Помощь 2Раз~рн 3Выход 4Hex 5Пер~ти 6 7Поиск 8Исх~ый 9Формат10Выход
```

Рис. 3.8: Файл содержащий текст программы

Оттранслировали текст программы lab6-1.asm в объектный файл. Выполнили компоновку объектного файла и запустили получившийся исполняемый файл. Программа вывела строку 'Введите строку:' и ожидала ввода с клавиатуры. (рис. 3.9)



```
elmdiago@fedora lab06]$ nasm -f elf lab6-1.asm
elmdiago@fedora lab06]$ ld -m elf_i386 -o lab6-1 lab6-1.o
elmdiago@fedora lab06]$ ./lab6-1
Введите строку:
Eliseu
elmdiago@fedora lab06]$
```

Рис. 3.9: Получившийся исполняемый файл

Скачали файл in\_out.asm со страницы курса в ТУИС. (рис. 3.10)

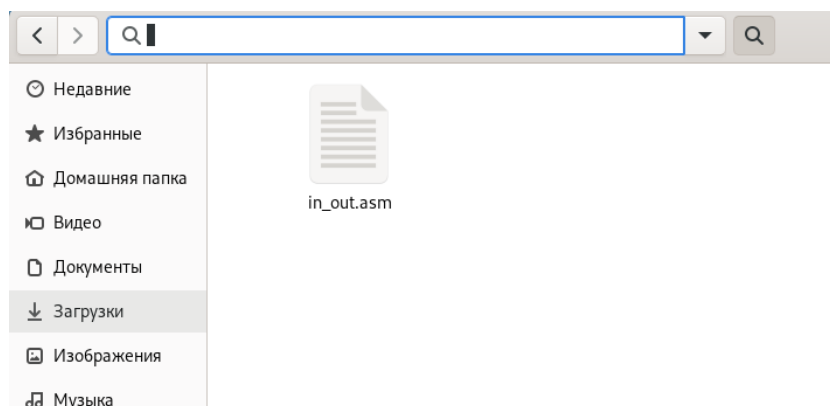


Рис. 3.10: Скачанный файл in\_out.asm

Подключаемый файл in\_out.asm должен лежать в том же каталоге, что и файл с программой, в которой он используется. В одной из панелей tc открыли каталог с файлом lab6-1.asm. В другой панели каталог со скачанным файлом in\_out.asm. Скопировали файл in\_out.asm в каталог с файлом lab6-1.asm с помощью функциональной клавиши F5.(рис. 3.11)

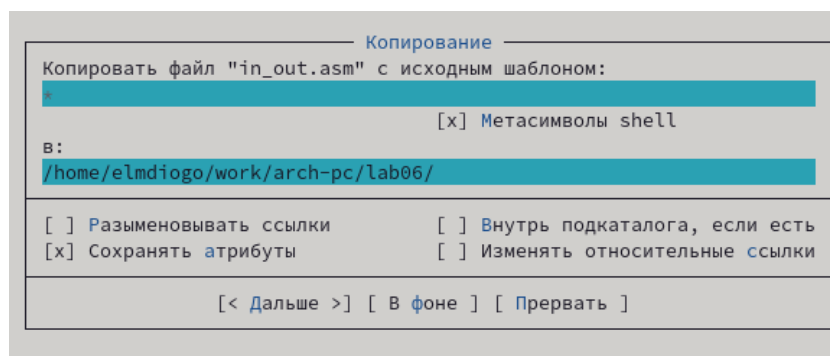


Рис. 3.11: Копирование файла in\_out.asm

С помощью функциональной клавиши F6 создали копию файла lab6- 1.asm с именем lab6-2.asm. Выделили файл lab6-1.asm, нажали клавишу F6 , ввели имя файла lab6-2.asm и нажали клавишу Enter. (рис. 3.12)

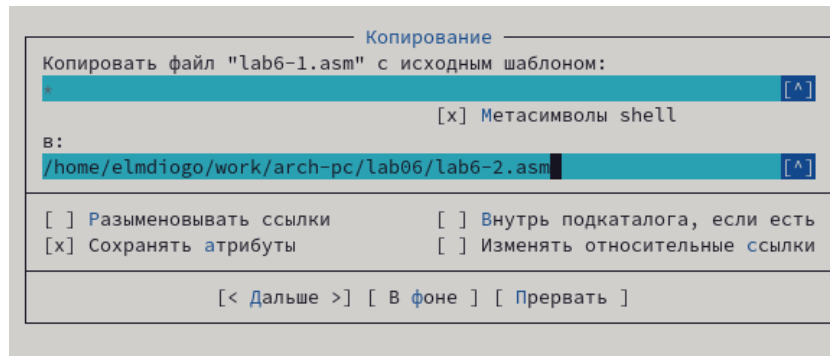


Рис. 3.12: lab6-2.asm

Исправили текст программы в файле lab6-2.asm с использованием подпрограмм из внешнего файла in\_out.asm (использовали подпрограммы sprintf, fread и quit) в соответствии с листингом 6.2. Создали исполняемый файл и проверьте его работу. (рис. 3.13), (рис. 3.14)

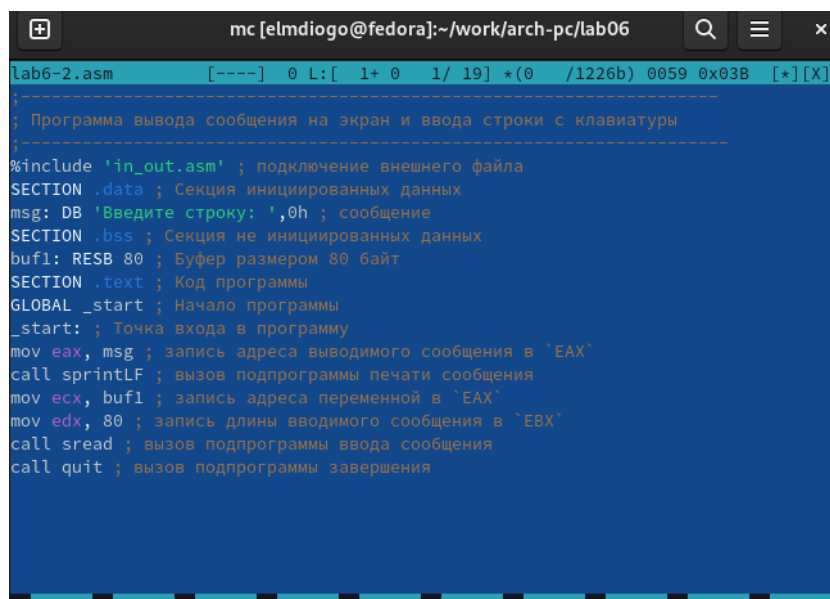


Рис. 3.13: sprintf, fread и quit

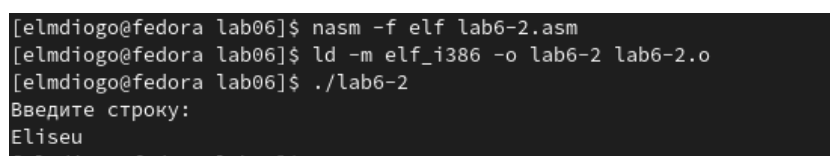


Рис. 3.14: Исполняемый файл

В файле lab6-2.asm заменили подпрограмму sprintLF на sprint. Создали исполняемый файл и проверили его работу. При замене подпрограмм пропадает перенос строки перед вводом текста. (рис. 3.15), (рис. 3.16)

```
call sprint ; вызов подпрограммы печати сообщения
```

Рис. 3.15: sprint

```
[elmdiago@fedora lab06]$ nasm -f elf lab6-2.asm
[elmdiago@fedora lab06]$ ld -m elf_i386 -o lab6-2 lab6-2.o
[elmdiago@fedora lab06]$ ./lab6-2
Введите строку: Eliseu
[elmdiago@fedora lab06]$
```

Рис. 3.16: Перенос строки перед вводом текста

## 3.2 Задание для самостоятельной работы

Создали копию файла lab6-1.asm. Внесли изменения в программу (без использования внешнего файла in\_out.asm), так чтобы она работала по следующему алгоритму: 1. вывести приглашение типа “Введите строку:”; 2. ввести строку с клавиатуры; 3. вывести введенную строку на экран.

(рис. 3.17), (рис. 3.18)

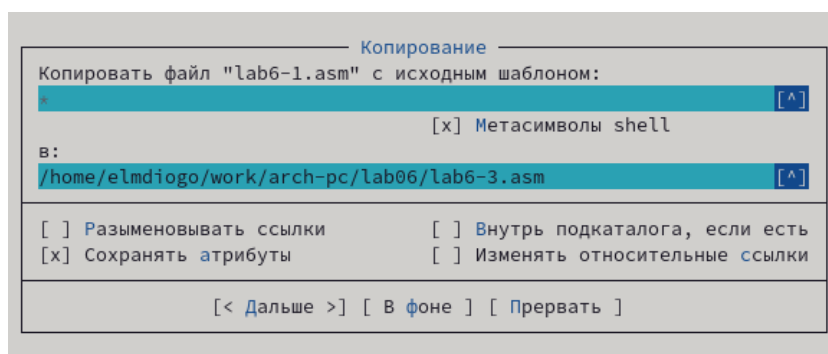


Рис. 3.17: lab6-3.asm

```

mc [elmdiago@fedora]:~/work/arch-pc/lab06
lab6-3.asm [----] 0 L: [ 1+ 0 1/ 39] *(0 /2335b) 0010 0x00A [*][X]

SECTION .data ; Секция инициализированных данных
msg: DB 'Введите строку:',10 ; сообщение плюс
; символ перевода строки
msgLen: EQU $-msg ; Длина переменной 'msg'
SECTION .bss ; Секция не инициализированных данных
buf1: RESB 80 ; Буфер размером 80 байт
;----- Текст программы -----
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
;----- Системный вызов 'write' -----
; После вызова инструкции 'int 80h' на экран будет
; выведено сообщение из переменной 'msg' длиной 'msgLen'
mov eax,4 ; Системный вызов для записи (sys_write)
mov ebx,1 ; Описатель файла 1 - стандартный вывод
mov ecx,msg ; Адрес строки 'msg' в 'ecx'
mov edx,msgLen ; Размер строки 'msg' в 'edx'
int 80h ; Вызов ядра
;----- системный вызов 'read' -----
; После вызова инструкции 'int 80h' программа будет ожидать ввода
; строки, которая будет записана в переменную 'buf1' размером 80 байт
;----- Меню -----
1Помощь 2Сох-ть 3Блок 4Замена 5Копия 6Пер-ть 7Поиск 8Уда-ть 9Меню 10Выход

```

Рис. 3.18: Изменения в программе

Получили исполняемый файл и проверили его работу. На приглашение ввести строку ввели свою фамилию. (рис. 3.19)

```

[elmdiago@fedora lab06]$ nasm -f elf lab6-3.asm
[elmdiago@fedora lab06]$ ld -m elf_i386 -o lab6-3 lab6-3.o
[elmdiago@fedora lab06]$ ./lab6-3
Введите строку:
Diogo
Diogo
[elmdiago@fedora lab06]$

```

Рис. 3.19: Проверка работы файла

Создали копию файла lab6-2.asm. Исправили текст программы с использованием подпрограмм из внешнего файла in\_out.asm, так чтобы она работала по следующему алгоритму: 1. вывести приглашение типа “Введите строку:”; 2. ввести строку с клавиатуры; 3. вывести введенную строку на экран. (рис. 3.20), (рис. 3.21)

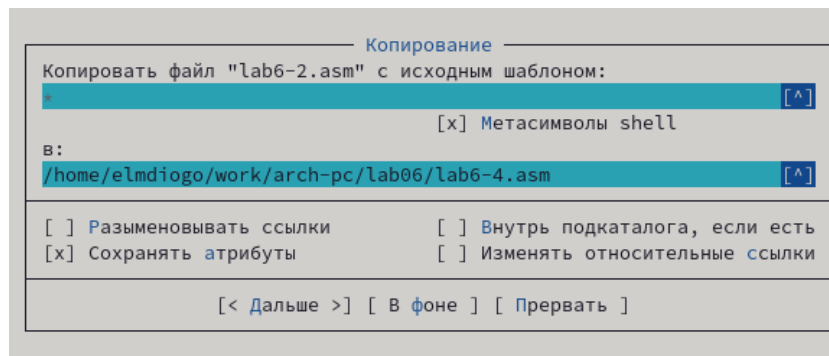


Рис. 3.20: lab6-4.asm

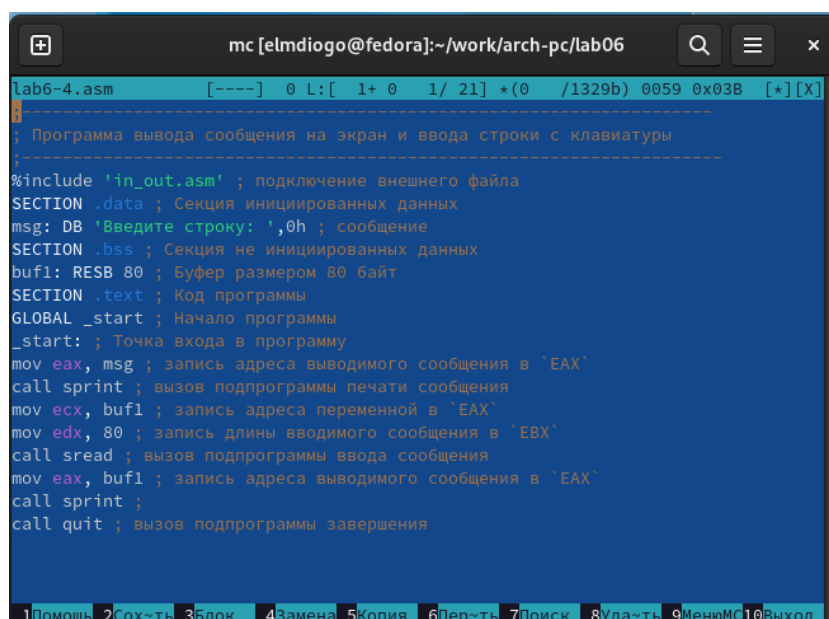


Рис. 3.21: Использование подпрограмм из внешнего файла in\_out.asm

Создали исполняемый файл и проверили его работу. (рис. 3.22)

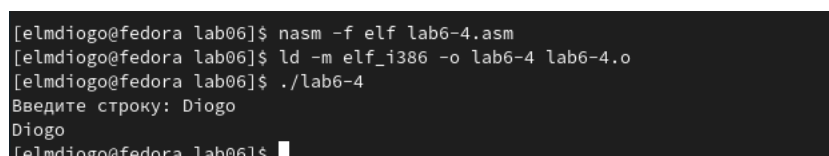


Рис. 3.22: Проверка работы исполняемого файла

## 4 Выводы

В ходе выполнения лабораторной работы были приобретены практически навыки работы в Midnight Commander и освоены инструкции языка ассемблера `mov` и `int`.