

Шаблон отчёта по лабораторной работе №6

НММбд-03-22

Толстых Максим Алексеевич

Содержание

1	Цель работы	5
2	Задание	6
3	Выполнение лабораторной работы	7
4	Задание для самостоятельной работы	16
5	Выводы	19

Список иллюстраций

3.1	Команда для вызова МС	7
3.2	Каталог arch-pc в МС	8
3.3	lab06	8
3.4	Команда touch	9
3.5	lab6-1.asm	9
3.6	Файл lab6-1.asm для редактирования	10
3.7	Текст программы из листинга 6.1	10
3.8	Файл содержащий текст программы	11
3.9	Получившийся исполняемый файл	11
3.10	Скачанный файл in_out.asm	12
3.11	Копирование файла in_out.asm	12
3.12	lab6-2.asm	13
3.13	sprintLF, sread и quit	14
3.14	Исполняемый файл	14
4.1	lab6-3.asm	16
4.2	Изменения в программе	17
4.3	Проверка работы файла	17
4.4	lab6-4.asm	18
4.5	Использование подпрограмм из внешнего файла in_out.asm	18
4.6	Проверка работы исполняемого файла	18

Список таблиц

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

Открываем Midnight Commander (рис. 1)

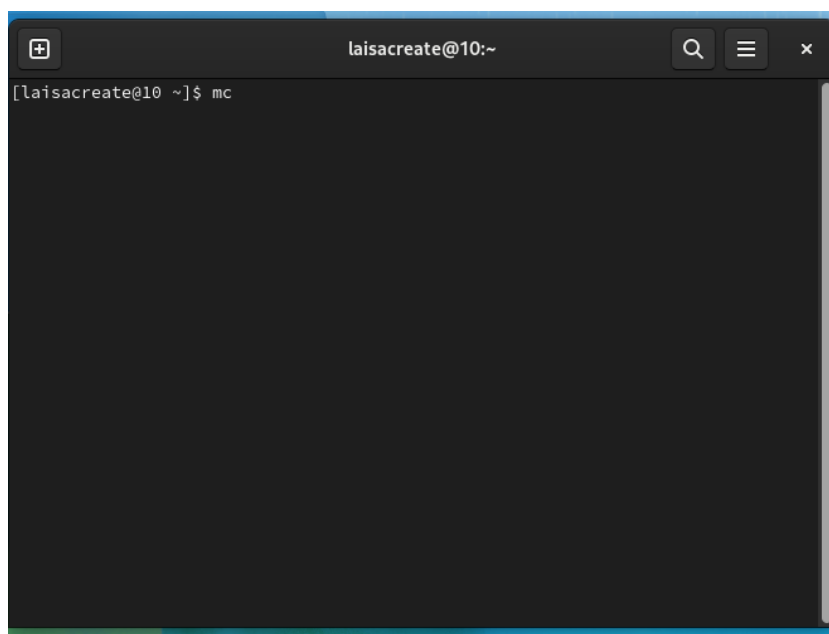


Рис. 3.1: Команда для вызова МС

Пользуясь клавишами “стрелочка вверх”, “стрелочка вниз” и Enter перешли в каталог arch-рс, созданный при выполнении лабораторной работы №5. (рис. 2)

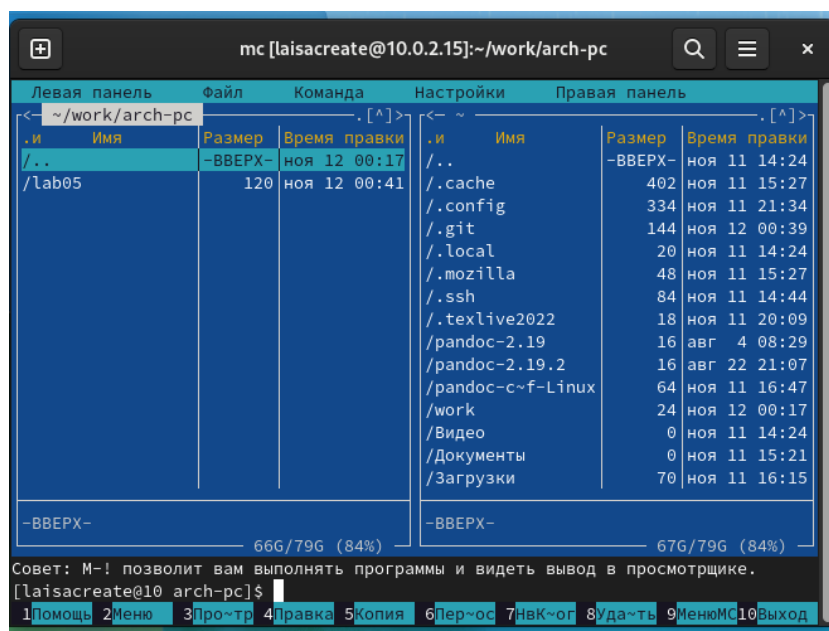


Рис. 3.2: Каталог arch-pc в MC

С помощью функциональной клавиши F7 создали папку lab06 и перешли в созданный каталог. (рис. 3)

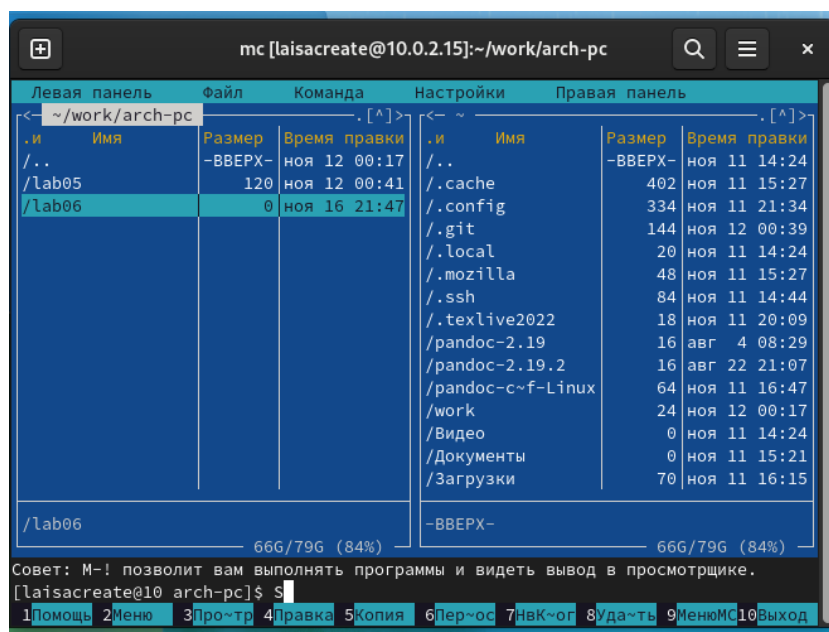


Рис. 3.3: lab06

Пользуясь строкой ввода и командой touch создайте файл lab6-1.asm. (рис. 4),

(рис. 5)

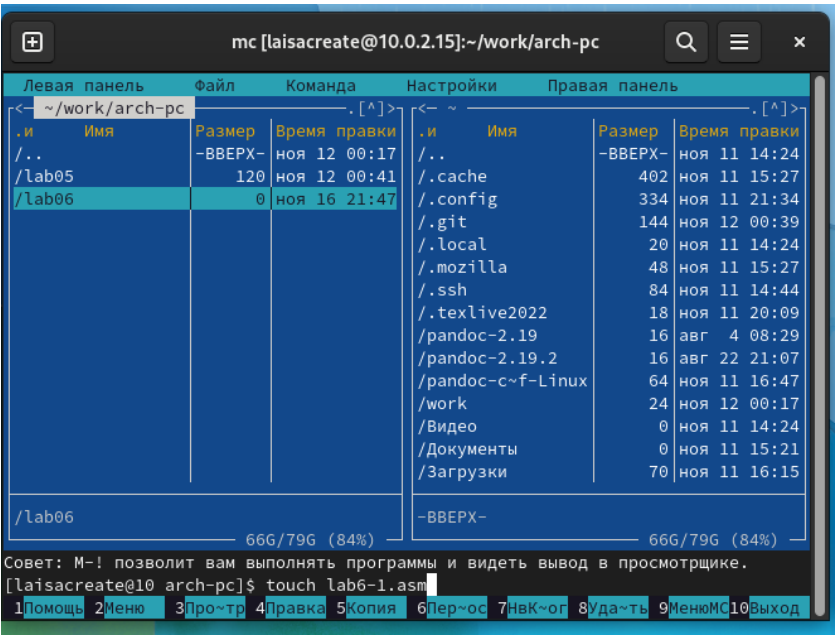


Рис. 3.4: Команда touch

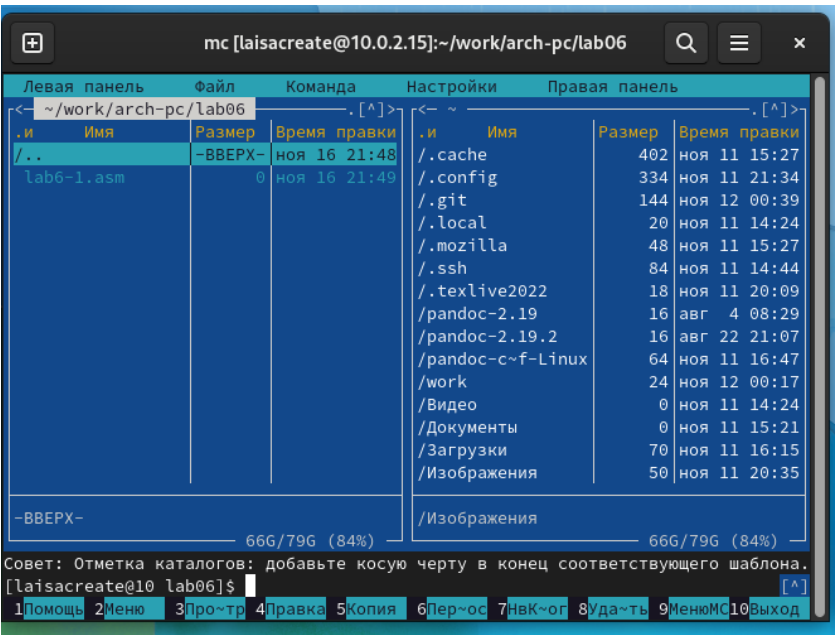


Рис. 3.5: lab6-1.asm

С помощью функциональной клавиши F4 откройте файл lab6-1.asm для редактирования во встроенном редакторе. (рис. 6)

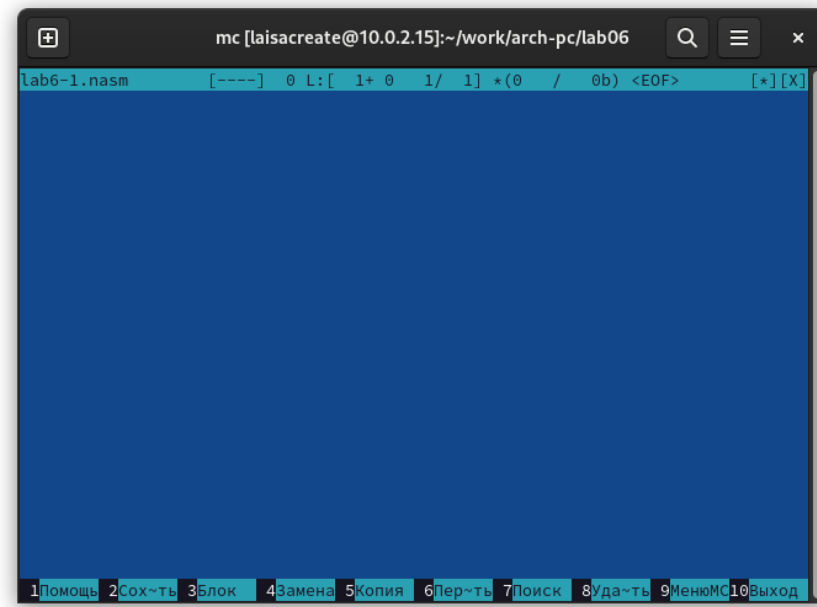


Рис. 3.6: Файл lab6-1.asm для редактирования

Ввели текст программы из листинга 6.1, сохранили изменения и закрыли файл.
(рис. 7)

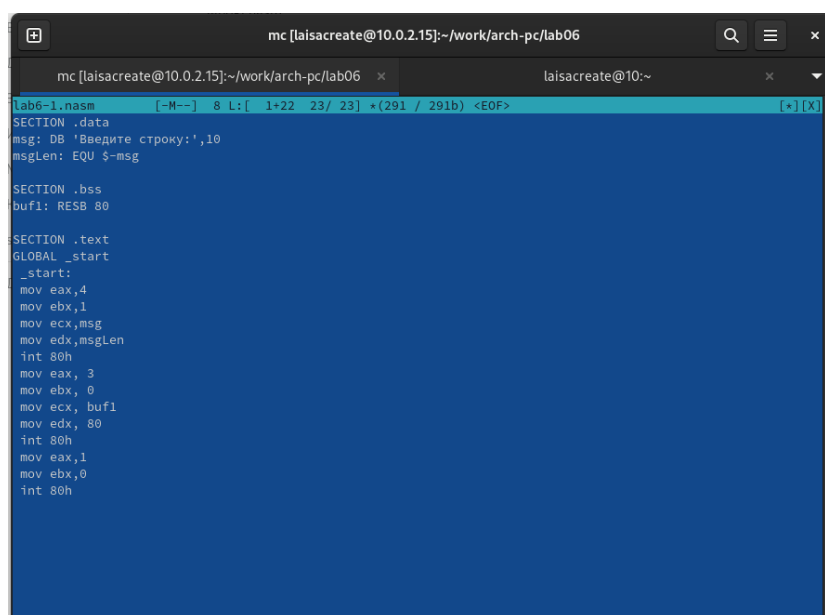
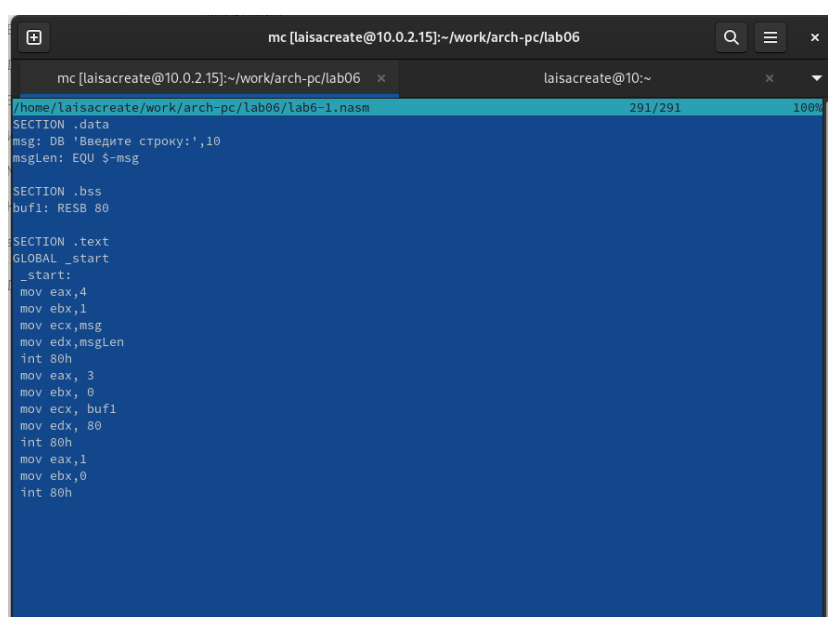


Рис. 3.7: Текст программы из листинга 6.1

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

смотрим. Убедились, что файл содержит текст программы. (рис. 8)



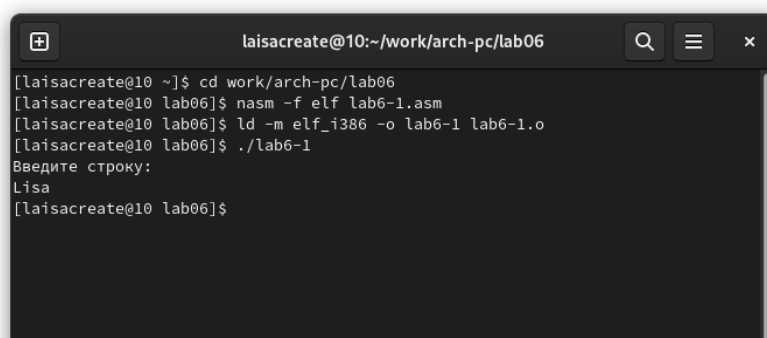
```
mc [laisacreate@10.0.2.15]:~/work/arch-pc/lab06
/home/laisacreate/work/arch-pc/lab06/lab6-1.nasm 291/291 100%
SECTION .data
msg: DB 'Введите строку:',10
msgLen: EQU $-msg

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
```

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. На запрос ввели Lisa. (рис. 9)



```
laisacreate@10:~/work/arch-pc/lab06
[laisacreate@10 ~]$ cd work/arch-pc/lab06
[laisacreate@10 lab06]$ nasm -f elf lab6-1.asm
[laisacreate@10 lab06]$ ld -m elf_i386 -o lab6-1 lab6-1.o
[laisacreate@10 lab06]$ ./lab6-1
Введите строку:
Lisa
[laisacreate@10 lab06]$
```

Рис. 3.9: Получившийся исполняемый файл

Скачали файл in_out.asm со страницы курса в ТУИС. (рис. 10)

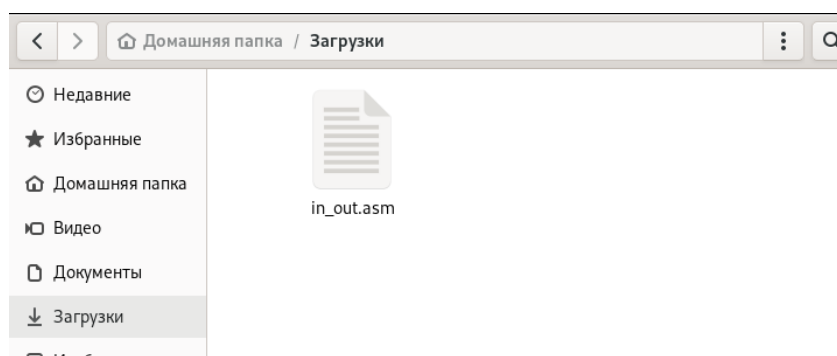


Рис. 3.10: Скачанный файл in_out.asm

Подключаемый файл in_out.asm должен лежать в том же каталоге, что и файл с программой, в которой он используется. В одной из панелей mc 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. 11)

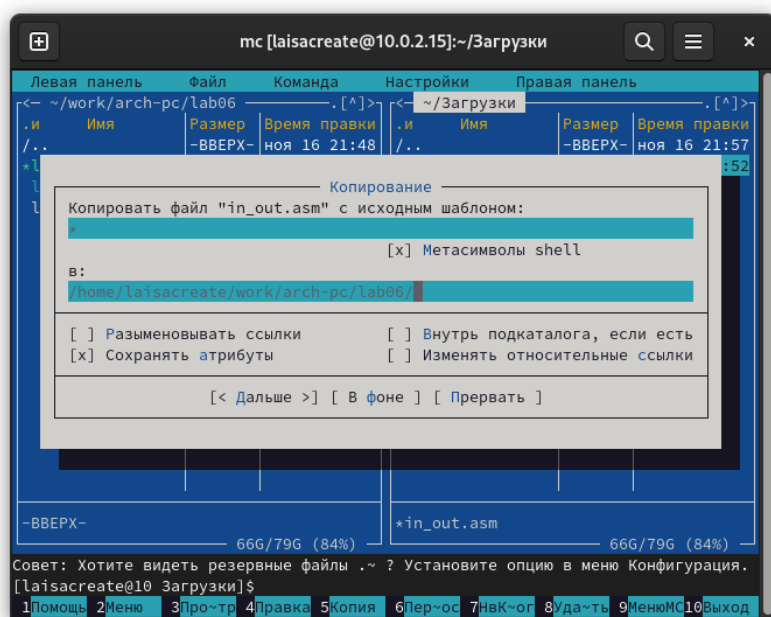


Рис. 3.11: Копирование файла in_out.asm

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. 12)

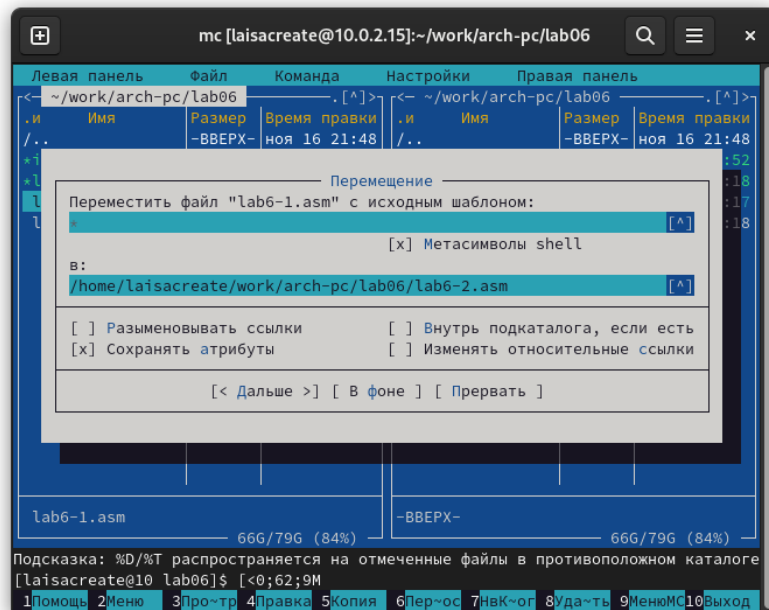
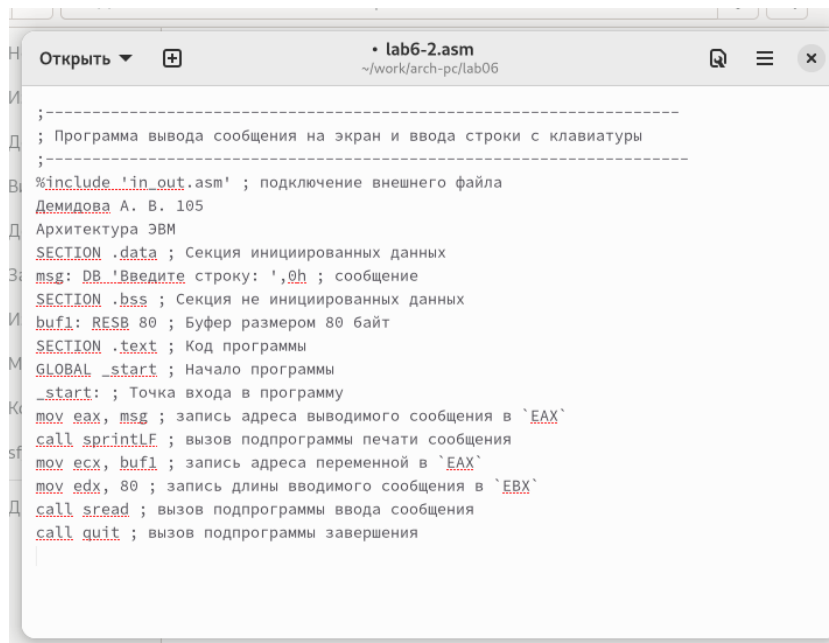


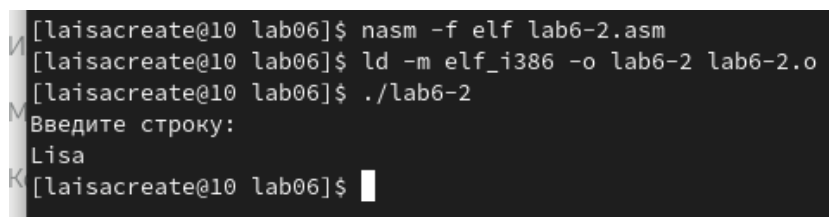
Рис. 3.12: lab6-2.asm

Исправили текст программы в файле lab6-2.asm с использованием подпрограмм из внешнего файла in_out.asm (использовали подпрограммы sprintLF, sread и quit) в соответствии с листингом 6.2. Создали исполняемый файл и проверьте его работу. (рис. 13), (рис. 14)



```
Открыть + lab6-2.asm ~/work/arch-pc/lab06
;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
%include 'in_out.asm' ; подключение внешнего файла
Демидова А. В. 105
Архитектура ЭВМ
SECTION .data ; Секция инициированных данных
msg: DB 'Введите строку: ',0h ; сообщение
SECTION .bss ; Секция не инициированных данных
buf1: RESB 80 ; Буфер размером 80 байт
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
mov eax, msg ; запись адреса выводимого сообщения в `EAX`
call sprintLF ; вызов подпрограммы печати сообщения
mov ecx, buf1 ; запись адреса переменной в `EAX`
mov edx, 80 ; запись длины вводимого сообщения в `EBX`
call sread ; вызов подпрограммы ввода сообщения
call quit ; вызов подпрограммы завершения
```

Рис. 3.13: sprintLF, sread и quit



```
[laisacrete@10 lab06]$ nasm -f elf lab6-2.asm
[laisacrete@10 lab06]$ ld -m elf_i386 -o lab6-2 lab6-2.o
[laisacrete@10 lab06]$ ./lab6-2
Введите строку:
Lisa
[laisacrete@10 lab06]$
```

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. 15), (рис. 16)

```

;-----
; Программа вывода сообщения на экран и ввода строки с клавиатуры
;-----
%include 'in_out.asm' ; подключение внешнего файла
SECTION .data ; Секция инициированных данных
msg: DB 'Введите строку: ',0h ; сообщение
SECTION .bss ; Секция не инициированных данных
buf1: RESB 80 ; Буфер размером 80 байт
SECTION .text ; Код программы
GLOBAL _start ; Начало программы
_start: ; Точка входа в программу
mov eax, msg ; запись адреса выводимого сообщения в `EAX`
call sprint ; вызов подпрограммы печати сообщения
mov ecx, buf1 ; запись адреса переменной в `EAX`
mov edx, 80 ; запись длины вводимого сообщения в `EBX`
call sread ; вызов подпрограммы ввода сообщения
call quit ; вызов подпрограммы завершения

```

```

[laiscreate@10 lab06]$ nasm -f elf lab6-2.asm
[laiscreate@10 lab06]$ ld -m elf_i386 -o lab6-2
[laiscreate@10 lab06]$ ./lab6-2
Введите строку: Lisa
[laiscreate@10 lab06]$

```

4 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. 17), (рис. 18)

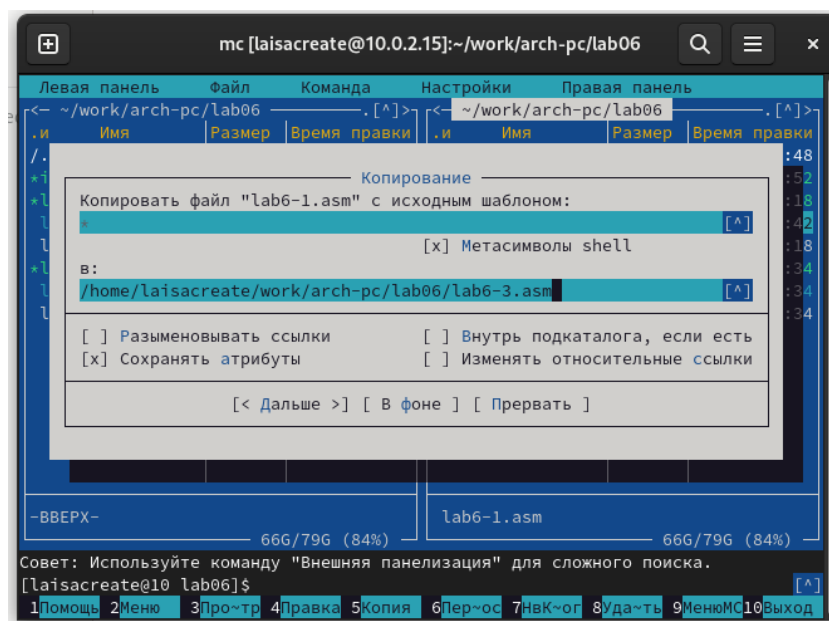


Рис. 4.1: lab6-3.asm


```

Открыть + lab6-3.asm ~/work/arch-pc/lab06

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
mov eax, 3 ; Системный вызов для чтения (sys_read)
mov ebx, 0 ; Дескриптор файла 0 - стандартный ввод
mov ecx, buf1 ; Адрес буфера под вводимую строку
mov edx, 80 ; Длина вводимой строки
int 80h ; Вызов ядра

mov eax,4
mov ebx,1
mov ecx, buf1
mov edx, 80
int 80h
;----- Системный вызов `exit` -----
; После вызова инструкции 'int 80h' программа завершит работу
mov eax,1 ; Системный вызов для выхода (sys_exit)
mov ebx,0 ; Выход с кодом возврата 0 (без ошибок)
int 80h ; Вызов ядра

```

Рис. 4.2: Изменения в программе

Получили исполняемый файл и проверили его работу. На приглашение ввести строку ввели свою фамилию. (рис. 19)

```

laisacreate@10:~/work/arch-pc/lab06
mc [laisacreate@10.0.2.15]:~/wor... x laisacreate@10:~/work/arch-pc/la... x
[laisacreate@10 ~]$ cd work/arch-pc/lab06
[laisacreate@10 lab06]$ nasm -f elf lab6-3.asm
[laisacreate@10 lab06]$ ld -m elf_i386 -o lab6-3 lab6-3.o
[laisacreate@10 lab06]$ ./lab6-3
Введите строку:
Popova
Popova
[laisacreate@10 lab06]$

```

Рис. 4.3: 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. 20), (рис. 21)

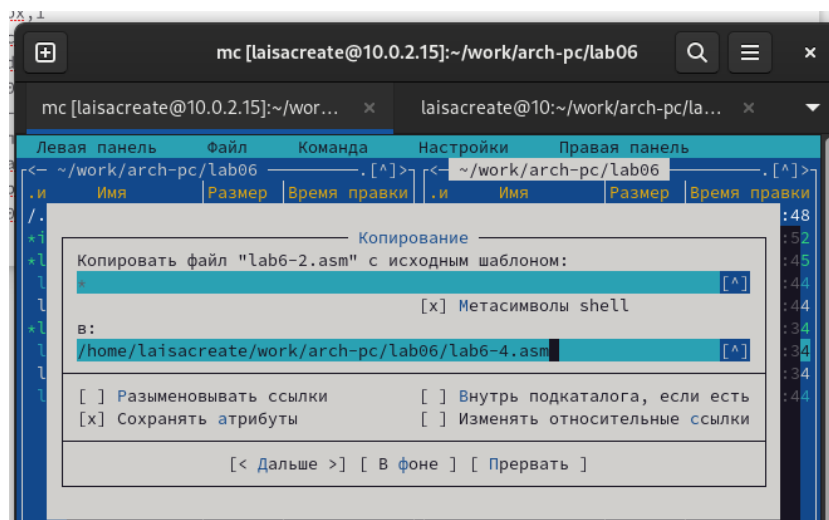
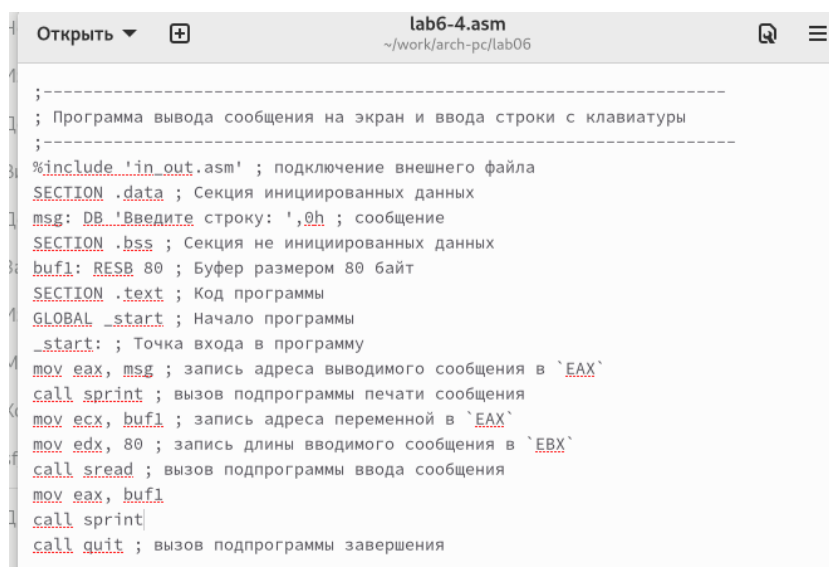
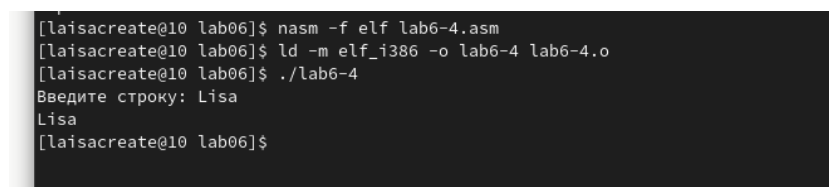


Рис. 4.4: lab6-4.asm



Создали исполняемый файл и проверили его работу. (рис. 22)



5 Выводы

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