

# **Отчёта по лабораторной работе 6**

**Основы работы с Midnight Commander (mc). Структура программы на языке ассемблера NASM**

Негин Голчин Задех

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Задание</b>	<b>6</b>
<b>3</b>	<b>Теоретическое введение</b>	<b>7</b>
<b>4</b>	<b>Выполнение лабораторной работы</b>	<b>8</b>
<b>5</b>	<b>Выводы</b>	<b>18</b>
	<b>Список литературы</b>	<b>19</b>

## Список иллюстраций

4.1	Создание файлов в Midnight Commander . . . . .	8
4.2	Редактирование файла 1 в Midnight Commander . . . . .	9
4.3	Проверка программы 1 . . . . .	10
4.4	Файл in_out.asm . . . . .	11
4.5	Редактирование файла 2 в Midnight Commander . . . . .	12
4.6	Проверка программы 2 . . . . .	13
4.7	Редактирование файла 3 в Midnight Commander . . . . .	14
4.8	Проверка программы 3 . . . . .	14
4.9	Редактирование файла 4 в Midnight Commander . . . . .	15
4.10	Проверка программы 4 . . . . .	16
4.11	Редактирование файла 5 в Midnight Commander . . . . .	17
4.12	Проверка программы 5 . . . . .	17

## **Список таблиц**

# 1 Цель работы

Целью работы является приобретение практических навыков работы в Midnight Commander. Освоение инструкций языка ассемблера `mov` и `int`.

## 2 Задание

1. Изучите как работать в Midnight Commander.
2. Изучите примеры программ из задания к работе.
3. Дополните примеры в соответствии с заданием.
4. Загрузите файлы на GitHub.

### 3 Теоретическое введение

Midnight Commander - это программа, предназначенная для просмотра содержимого каталогов и выполнения основных функций управления файлами в UNIX-подобных операционных системах.

Главное окно программы Midnight Commander состоит из трех полей. Два поля, называемые “панелями”, идентичны по структуре и обычно отображают перечни файлов и подкаталогов каких-то двух каталогов файловой структуры. Эти каталоги в общем случае различны, хотя, в частности, могут и совпасть. Каждая панель состоит из заголовка, списка файлов и информационной строки.

Третье поле экрана, расположенное в нижней части экрана, содержит командную строку текущей оболочки. В этом же поле (самая нижняя строка экрана) содержится подсказка по использованию функциональных клавиш F1 - F10. Самая верхняя строка экрана содержит строку горизонтального меню.

Эта строка может не отображаться на экране; в этом случае доступ к ней можно получить, щелкнув мышью по верхней рамке или нажав клавишу F9.

Панели Midnight Commander обеспечивают просмотр одновременно двух каталогов. Одна из панелей является активной в том смысле, что пользователь может выполнять некоторые операции с отображаемыми в этой панели файлами и каталогами.

## 4 Выполнение лабораторной работы

1. Создадим новый подкаталог с именем lab06 и в нем файл lab6-1.asm. (рис. 4.1)

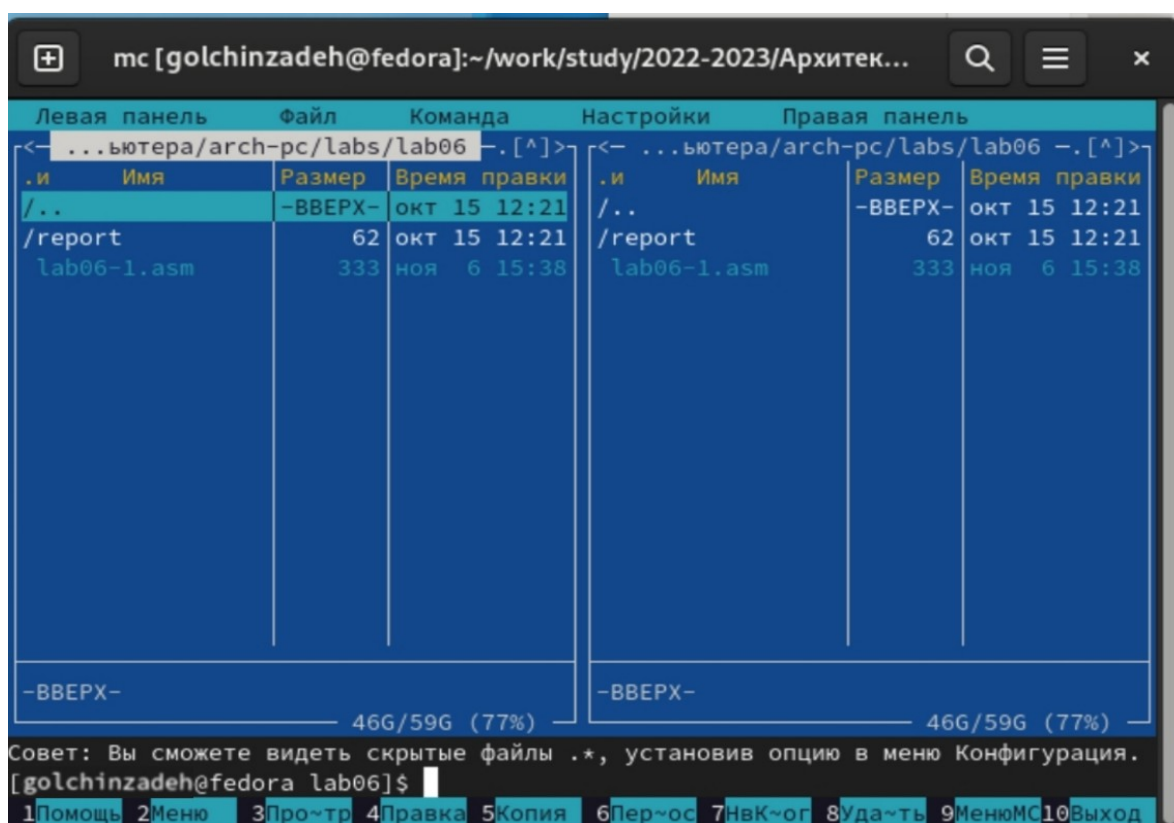
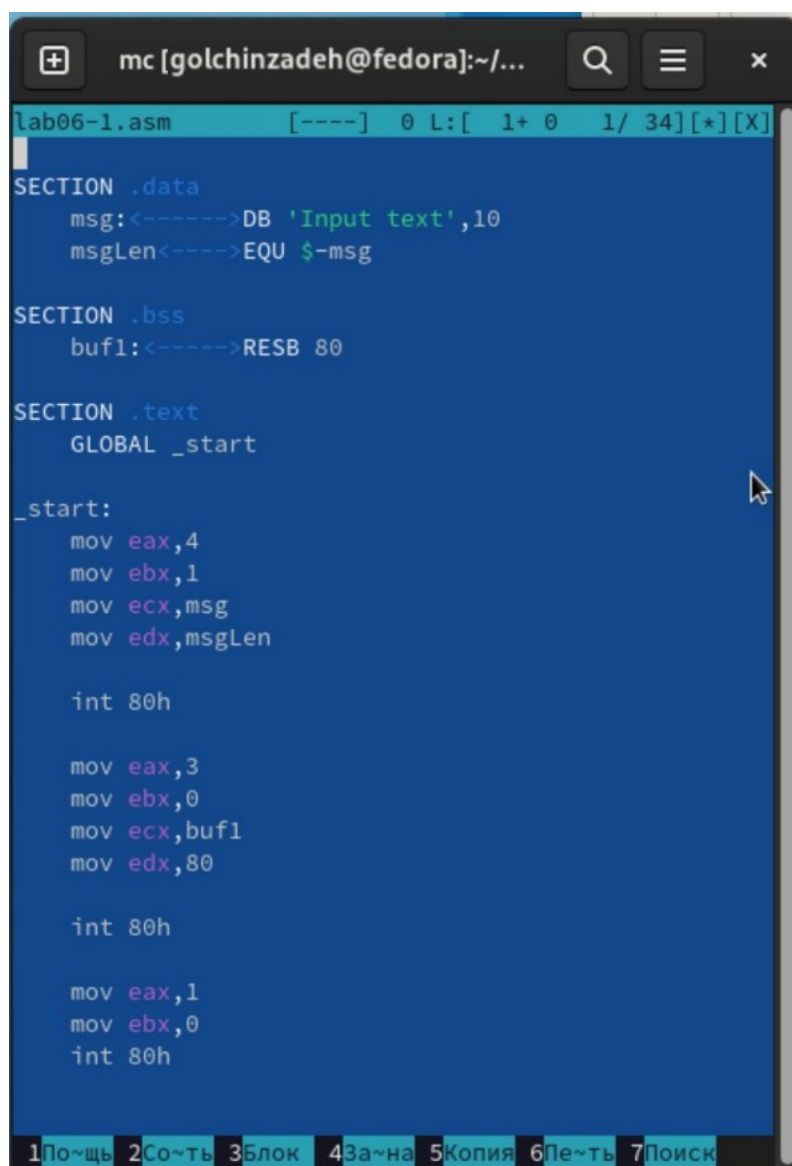


Рис. 4.1: Создание файлов в Midnight Commander

2. Введем в файл lab6-1.asm текст программы вывода сообщения на экран и ввода строки с клавиатуры (Листинг 1.). Создадим исполняемый файл и



проверим его работу. (рис. 4.2, 4.3)



```
lab06-1.asm [----] 0 L: [ 1+ 0 1/ 34] [*] [X]

SECTION .data
    msg:<----->DB 'Input text',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

1По~щъ 2Со~ть 3Блок 4За~на 5Копия 6Пе~ть 7Поиск
```

Рис. 4.2: Редактирование файла 1 в Midnight Commander

```
[golchinzadeh@fedora lab06]$ nasm -f elf lab06-1.asm
[golchinzadeh@fedora lab06]$ ld -o lab06-1 lab06-1.o
ld: архитектура i386 входного файла «lab06-1.o» несовместима с выходным i386:x86-64
[golchinzadeh@fedora lab06]$ ld -o -m elf_i386 lab06-1 lab06-1.o
ld: невозможно найти elf_i386: Нет такого файла или каталога
ld: невозможно найти lab06-1: Нет такого файла или каталога
[golchinzadeh@fedora lab06]$ ld -m elf_i386 -o lab06-1 lab06-1.o
[golchinzadeh@fedora lab06]$ ./lab06-1
Input text
Negin
```

Рис. 4.3: Проверка программы 1

3. Скачали с туис доп файл, скопировали программу. (рис. 4.4)

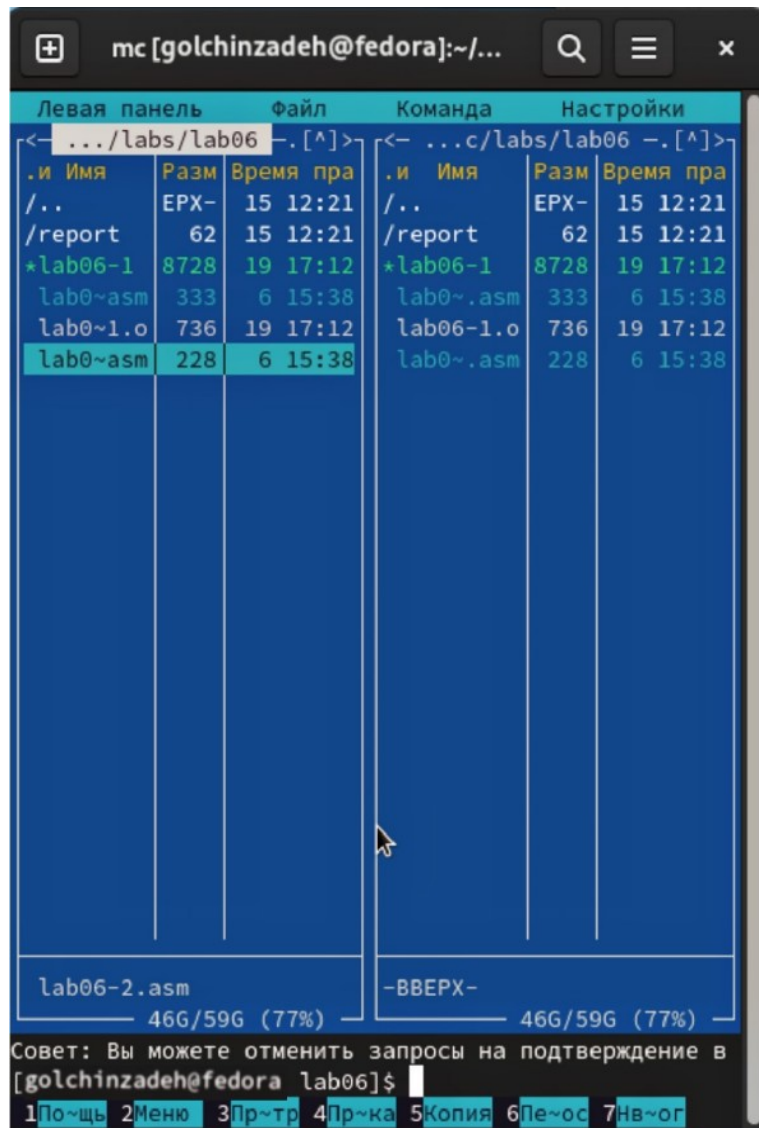


Рис. 4.4: Файл in\_out.asm

4. Изменили код программы. (рис. 4.5, 4.6)

```
lab06-2.asm      [----]  0 L:[ 1+16 17/ 17] [*] [X]
%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
```

Рис. 4.5: Редактирование файла 2 в Midnight Commander

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

Рис. 4.6: Проверка программы 2

5. Изменили вызов подпрограммы. Теперь ввод и вывод в одну строку. (рис. 4.7, 4.8)

```
lab06-3.asm [----] 17 L:[ 1+15 16/ 17][*][X]
#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
```

Рис. 4.7: Редактирование файла 3 в Midnight Commander

```
[golchinzadeh@fedora lab06]$ nasm -f elf lab06-3.asm
[golchinzadeh@fedora lab06]$ ld -m elf_i386 -o lab06-3 lab06-3.o
[golchinzadeh@fedora lab06]$ ./lab06-3
Введите строку: Negin
```

Рис. 4.8: Проверка программы 3

6. Внесем изменения в программу (без использования внешнего файла in\_out.asm), так чтобы она работала по следующему алгоритму: (рис. 4.9, 4.10)

- вывести приглашение типа “Введите строку:”;
- ввести строку с клавиатуры;
- вывести введенную строку на экран.



```
lab06-4.asm [----] 0 L: [ 1+39 40/ 40] [*] [X]

SECTION .data
    msg:<----->DB 'Input text',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
```

Рис. 4.9: Редактирование файла 4 в Midnight Commander

```
[golchinzadeh@fedora lab06]$ nasm -f elf lab06-4.asm
[golchinzadeh@fedora lab06]$ ld -m elf_i386 -o lab06-4 lab06-4.o
[golchinzadeh@fedora lab06]$ ./lab06-4
Input text
Negin
Negin
```

Рис. 4.10: Проверка программы 4

7. Сделаем тоже самое с файлом in\_out.asm (рис. 4.11, 4.12)



```
lab06-5.asm  [----]  0 L:[ 1+21 22/ 22][*][X]
%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 sprintLF
    .
<----->call quit
```

Рис. 4.11: Редактирование файла 5 в Midnight Commander

```
[golchinzadeh@fedora lab06]$ nasm -f elf lab06-5.asm
[golchinzadeh@fedora lab06]$ ld -m elf_i386 -o lab06-5 lab06-5.o
[golchinzadeh@fedora lab06]$ ./lab06-5
Введите строку: Negin
Negin
```

Рис. 4.12: Проверка программы 5

## 5 Выводы

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

# Список литературы

1. Справочная система по языку Assembler
2. Midnight Commander