

# **. Отчёт по лабораторной работе №5**

**Основы работы с Midnight Commander**

Хохлачёва Полина Дмитриевна

# Содержание

<b>1</b>	<b>Цель работы</b>	<b>5</b>
<b>2</b>	<b>Выполнение лабораторной работы</b>	<b>6</b>
<b>3</b>	<b>Выводы</b>	<b>15</b>

## Список иллюстраций

2.1	Вводим в консоле команду <code>mc</code> . . . . .	6
2.2	Создаём с помощью <code>F7</code> . . . . .	7
2.3	Создаём с помощью команды <code>touch</code> . . . . .	7
2.4	Открываем файл, заполняем и сохраняем . . . . .	8
2.5	Открываем файл и убеждаемся, что файл содержит текст . . . . .	9
2.6	Проверяем работу . . . . .	9
2.7	Скачиваем . . . . .	10
2.8	Создаём копию . . . . .	10
2.9	С помощью клавиши <code>F6</code> . . . . .	11
2.10	Открываем файл, заполняем и сохраняем . . . . .	11
2.11	Создаём копию . . . . .	12
2.12	Копируем файл . . . . .	12
2.13	Редактируем . . . . .	13
2.14	Копируем файл . . . . .	13
2.15	Копируем файл . . . . .	14

## **Список таблиц**

# 1 Цель работы

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

|

## 2 Выполнение лабораторной работы

Описываются проведённые действия, в качестве иллюстрации даётся ссылка на иллюстрацию (рис. 2.1). Открываем Midnight Commander

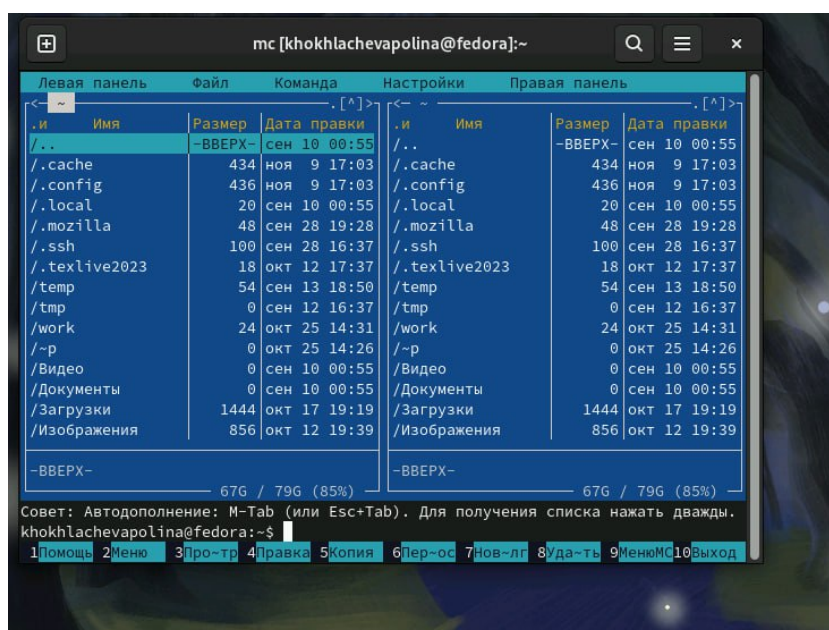
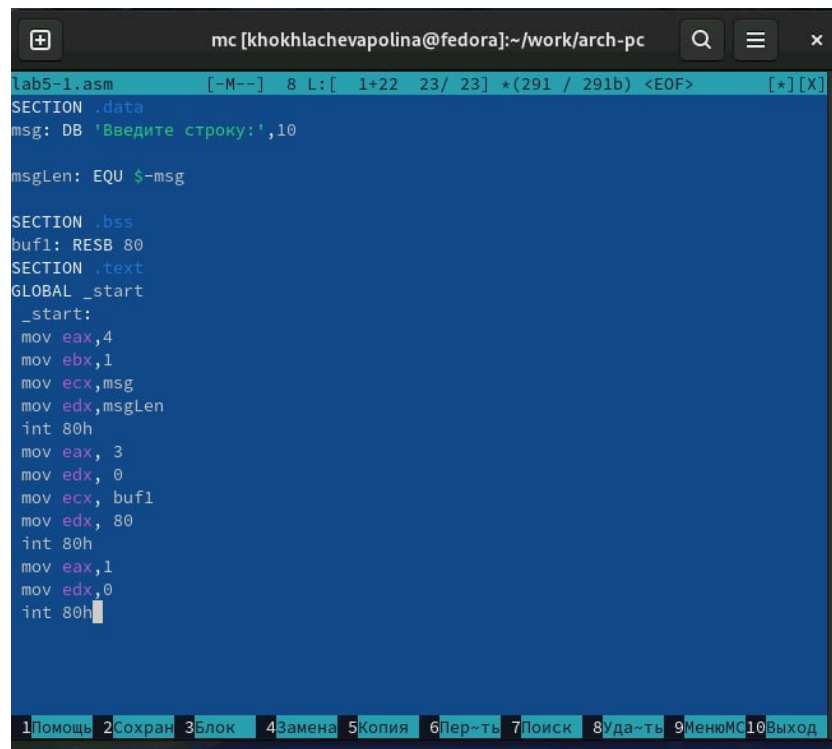


Рис. 2.1: Вводим в консоле команду mc

Создаём каталог lab05



Открываем файл для редактирования и заполнения по листингу



```
lab5-1.asm [-M--] 8 L: [ 1+22 23/ 23] *(291 / 291b) <EOF> [*] [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 edx, 0
mov ecx, buf1
mov edx, 80
int 80h
mov eax,1
mov edx,0
int 80h
```

1Помощь 2Сохран 3Блок 4Замена 5Копия 6Пер-ть 7Поиск 8Уда-ть 9МенюMC10Выход

Рис. 2.4: Открываем файл, заполняем и сохраняем

Открываем файл для просмотра



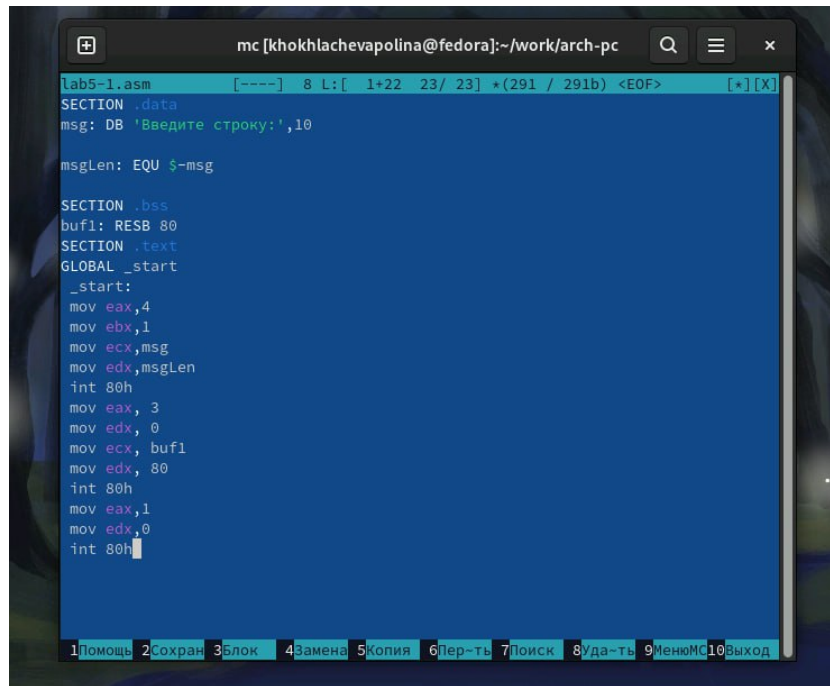


Рис. 2.5: Открываем файл и убеждаемся, что файл содержит текст

Транслируем текст программы и запускаем исполняемый файл

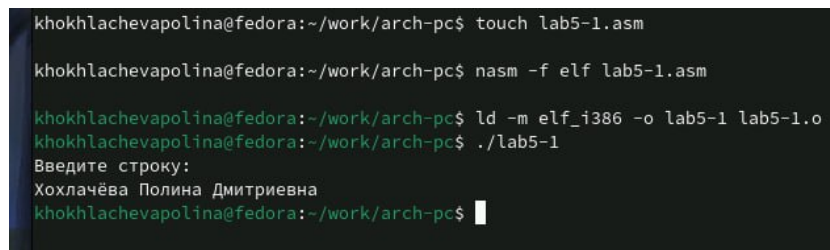


Рис. 2.6: Проверяем работу

Скачиваем файл

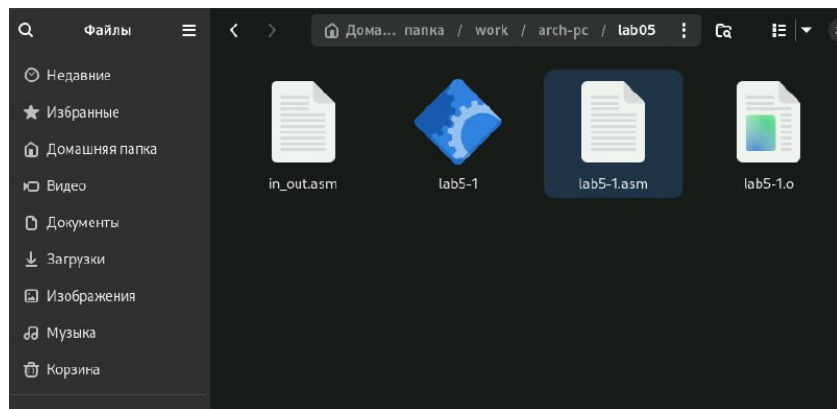


Рис. 2.7: Скачиваем

Копируем файл

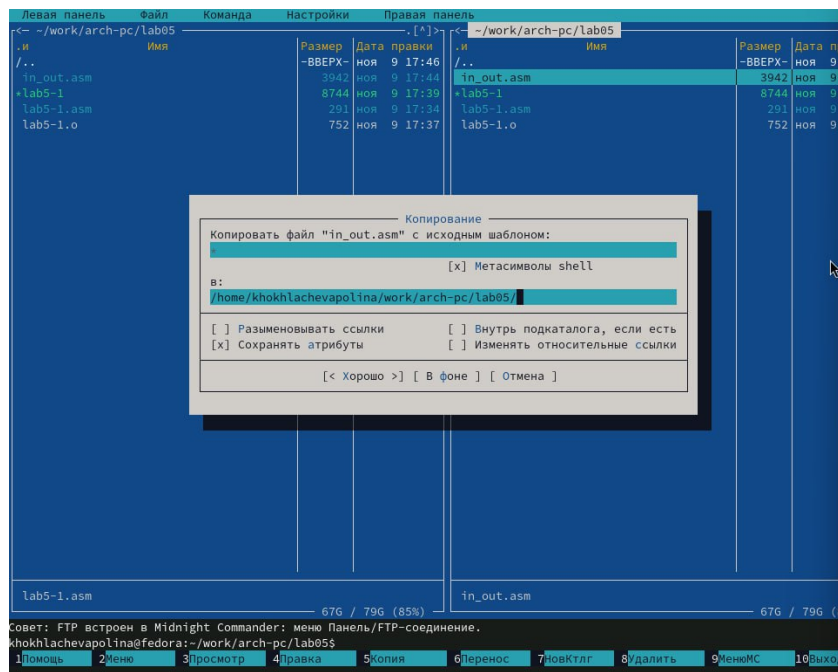


Рис. 2.8: Создаём копию

Создаём копию

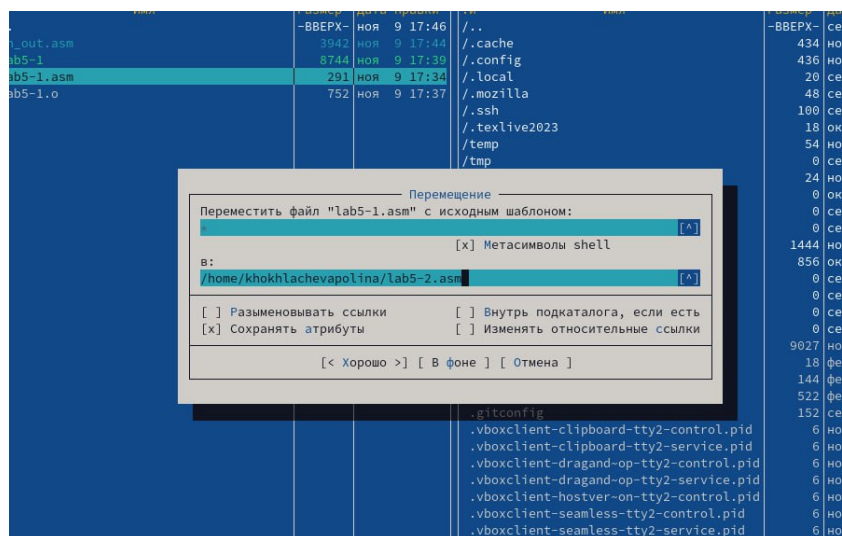


Рис. 2.9: С помощью клавиши F6

Новый файл и заполняем его в соответствии с листингом

```
lab5-2.asm [-M--] 13 L: [ 1+13 14/ 14] *(
%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 spread
call quit
```

Рис. 2.10: Открываем файл, заполняем и сохраняем

Снова копию файла lab5-1.asm

< ~/work/arch-pc/lab05		. [^]>	
Имя	Размер	Дата правки	
../	-ВВЕРХ-	ноя 9 17:46	
in_out.asm	3942	ноя 9 17:44	
*lab5-1	8744	ноя 9 17:39	
lab5-1.asm	292	ноя 9 18:20	
lab5-1.o	752	ноя 9 17:37	
lab5-2.asm	246	ноя 9 18:34	
lab5-3.asm	0	ноя 9 18:35	

Рис. 2.11: Создаём копию

Копируем файл

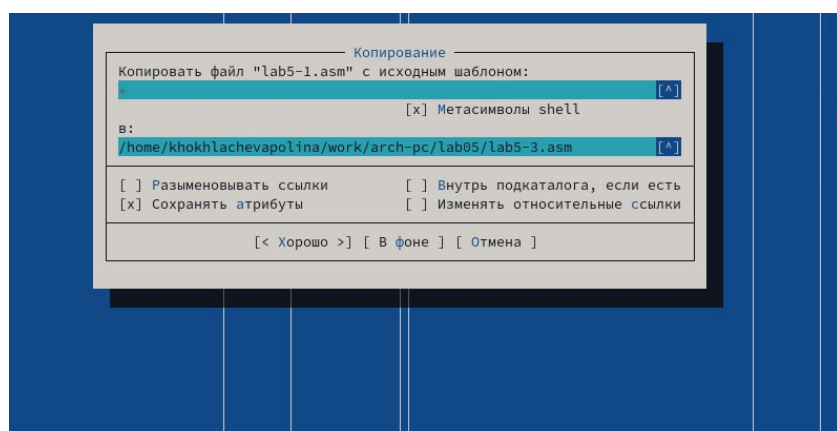


Рис. 2.12: Копируем файл

Редактируем файл

```

lab5-3.asm      [-M--]  8 L:[ 1+28  29/ 29] *(353 / 353b) <EOF>
SECTION data.
msg: DB 'Введите строку:',10
msgLen: EQU $-msg

SECTION .bss
buf1: RESB 80
SECTION .text
GLOBAL _start
_start:
mov eax,4
mov edx,1
mov ecx,msg
mov edx,msglen
int 80h
mov eax, 3
mov edx, 0
mov ecx, buf1
mov edx, 80
int 80h

mov eax, 4
mov edx, 1
mov ecx,buf1
mov edx,80
int 80h

mov eax,1
mov edx,0
int 80h

```

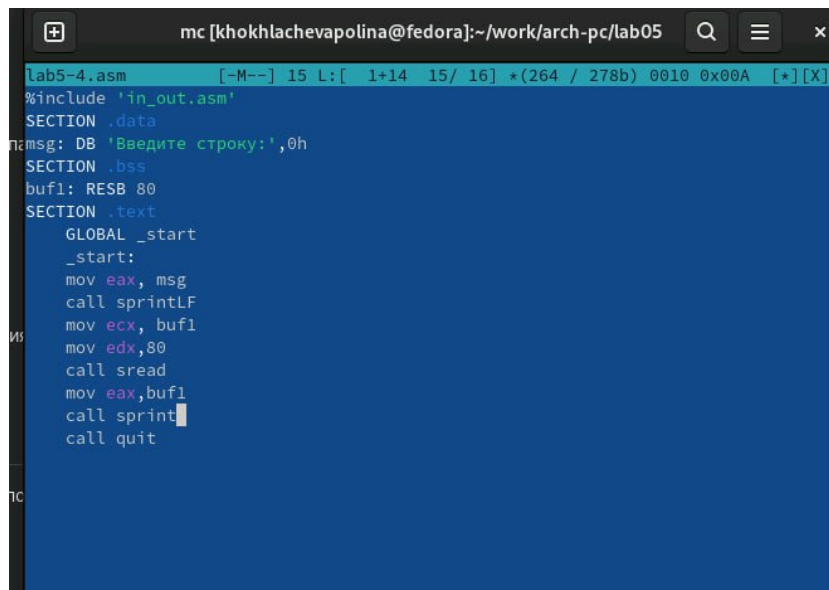
Рис. 2.13: Редактируем

Создаём копию файла lab5-2.asm

Левая панель			Правая панель		
Файл	Команда	Настройки	Файл	Команда	Настройки
./..	Имя	Размер	./..	Имя	Размер
in_out.asm	3942	ноя 9 17:44	in_out.asm	3942	ноя 9 17:44
*lab5-1	8744	ноя 9 17:39	*lab5-1	8744	ноя 9 17:39
lab5-1.asm	292	ноя 9 18:20	lab5-1.asm	292	ноя 9 18:20
lab5-2.asm	246	ноя 9 18:34	lab5-2.asm	246	ноя 9 18:34
lab5-3.asm	353	ноя 9 18:46	lab5-3.asm	353	ноя 9 18:46
lab5-4.asm	246	ноя 9 18:34	lab5-4.asm	246	ноя 9 18:34

Рис. 2.14: Копируем файл

Редактируем файл



```
lab5-4.asm [-M--] 15 L:[ 1+14 15/ 16] *(264 / 278b) 0010 0x00A [*][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
mov eax, buf1
call sprint
call quit
```

Рис. 2.15: Копируем файл

## 3 Выводы

Мы приобрели навыки работы в Midnight Commander и освоили инструкцию `mov`